## General Disclaimer

## One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

JUL 2 9 1981

# AgRISTARS

SR-L1-00304
JSC-17369

NASA-CR-161067

"Made available under NASA sponsorship in the interest of early and wide dissemination of Earth Resources Survey Program information and without liability for any use made thereof."

**A Joint Program for Agriculture and Resources Inventory Surveys Through Aerospace Remote Sensing**

## Supporting Research

June 1981

# "AS-BUILT" DESIGN SPECIFICATION FOR THE CLASFYG PROGRAM

E82-10092

C. L. Horton

(E82-10092) AS-BUILT DESIGN SPECIFICATION FOR THE CLASFYG PROGRAM (Lockheed Engineering and Management) 129 p
HC A07/MF A01
CSCL 02C
N82-21644
Unclas
G3/43 00092

Lockheed Engineering and Management Services Company, Inc.
1830 NASA Road 1, Houston, Texas 77058

**Lyndon B. Johnson Space Center**
Houston, Texas 77058

SR-L1-00304
JSC-17369

# "AS-BUILT" DESIGN SPECIFICATION FOR THE CLASFYG PROGRAM

Job Order 71-308

Prepared By

C. L. Horton

Approved By

G. L. Clouette, Supervisor
Support Systems Software Section

R. Kent Lennington, Supervisor
Techniques Development Section

R. A. McClane, Manager
Data Systems Department

T. C. Minter, Manager
Development and Evaluation Department

Prepared By

Lockheed Engineering and Management Services Company, Inc.

For

Earth Observations Division

Space and Life Sciences Directorate

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

June 1981

LEMSCO-16649

| 1. Report No. | 2. Government Accession No. | 3. Recipient's Catalog No. |
|---|---|---|
| JSC-17369, SR-L1-00304 | | |

| 4. Title and Subtitle | 5. Report Date |
|---|---|
| "As-Built" Design Specification for the CLASFYG Program | June 1981 |
| | 6. Performing Organization Code: SG3 |

| 7. Author(s) | 8. Performing Organization Report No. |
|---|---|
| C. L. Horton<br>Lockheed Engineering and Management Services Company, Inc. | LEMSCO-16649 |
| | 10. Work Unit No.: 63-2457-1308 |
| 9. Performing Organization Name and Address<br>Lockheed Engineering and Management Services Company, Inc.<br>1830 NASA Road 1<br>Houston, Texas 77058 | 11. Contract or Grant No.: NAS 9-15800 |
| | 13. Type of Report and Period Covered: Design Specification |
| 12. Sponsoring Agency Name and Address<br>National Aeronautics and Space Administration<br>Lyndon B. Johnson Space Center<br>Houston, Texas 77058 Dr. G. Badhwar / SG3 | 14. Sponsoring Agency Code: 626-48 |

15. Supplementary Notes

16. Abstract

This document is the "As-Built" Design Specification for the CLASFYG program. The program produces a file with a Universal-formatted header and data records in a non-standard format. Trajectory coefficients are calculated from 5 to 8 acquisitions of radiance values in the training field corresponding to an agricultural product. These coefficients are used to calculate a time of emergance and corresponding trajectory coefficients for each pixel in the test field. The time of emergance, two of the coefficients and the sigma value for each pixel are written to the file.

| 17. Key Words (Suggested by Author(s)) | 18. Distribution Statement |
|---|---|
| Universal-format<br>Trajectory coefficients<br>Classification | |

| 19. Security Classif. (of this report) | 20. Security Classif. (of this page) | 21. No. of Pages | 22. Price* |
|---|---|---|---|
| Unclassified | Unclassified | 129 | |

*For sale by the National Technical Information Service, Springfield, Virginia 22161

PRECEDING PAGE BLANK NOT FILMED

CONTENTS


| Section | | Page |
|---|---|---|
| 1. | SCOPE | 1-1 |
| 2. | APPLICABLE DOCUMENTS | 2-1 |
| 3. | SYSTEM DESCRIPTIONS | 3-1 |
| | 3.1 SYSTEM FLOWCHART | 3-1 |
| | 3.2 HARDWARE DESCRIPTION | 3-1 |
| | 3.3 SOFTWARE DESCRIPTION | 3-4 |
| | 3.4 FILE DESCRIPTION | 3-5 |
| | 3.4.1 USER INFORMATION FILE | 3-5 |
| | 3.4.2 ACQUISITION DATA FILES | 3-7 |
| | 3.4.3 DATA FROM RT&E DATA BASE | 3-8 |
| | 3.4.4 COEFFICIENTS FILE | 3-9 |
| | 3.5 DETAILED SOFTWARE DESCRIPTION | 3-11 |
| | 3.5.1 CLASFYG | 3-11 |
| | 3.5.2 CROPG | 3-14 |
| | 3.5.3 MOVSYM | 3-16 |
| | 3.5.4 NUMBFL | 3-18 |
| | 3.5.5 RDEXEC | 3-20 |
| | 3.5.6 RDFLES | 3-22 |
| | 3.5.7 RDVERT | 3-25 |
| | 3.5.8 RECT | 3-27 |
| | 3.5.9 SDEV | 3-29 |
| | 3.5.10 TRJXM | 3-31 |
| | 3.5.11 F2ERVG | 3-34 |

CONTENTS


| Section | | Page |
|---|---|---|
| 3.5.12 | FCHISQ | 3-36 |
| 3.5.13 | F2NCTG | 3-38 |
| 3.5.14 | MTINV1 | 3-40 |
| 3.5.15 | NUMBR | 3-42 |
| 4. | OPERATION INSTRUCTION | 4-1 |
| 4.1 | OPERATING DESCRIPTION | 4-1 |
| 4.2 | COMMANDS DESCRIPTION | 4-2 |
| 4.2.1 | START | 4-3 |
| 4.2.2 | CLASFYG | 4-4 |
| 4.2.3 | END | 4-5 |
| 4.3 | OPERATING EXAMPLE | 4-6 |

Appendices

| | | |
|---|---|---|
| A. | COMMON BLOCKS | A-1 |
| B. | CLASFYG CONSTANT DEFINITION | B-1 |
| C. | JOB CONTROL SOFTWARE | C-1 |
| D. | PROGRAM LISTINGS. | D-1 |
| E. | OUTPUT EXAMPLE | E-1 |

# CLASFYG PROGRAM

## 1.0 SCOPE

This document contains the description of the CLASFYG computer program. The program calculates coefficients to fit the model

$$G(t) = A \left(\frac{t}{t_o}\right)^{\alpha} e^{-\beta(t_o^2 - t^2)} \text{ for } t \geq t_o$$

and

$$G(t) = A \quad \text{for } t \leq t_o$$

where G is the greenness and A is the soil line, to the time trajectory of each pixel in a specified region of a segment. A disk file is written with a Universally formatted header record and line records recording the $\alpha$, $\beta$, $t_o$ and CHI-Square value for each pixel on the scan line.

## 2.0 APPLICABLE DOCUMENTS

The following documents form a part of this specification: AD 63-2457-3308-01 Transferring of Badhwar Software.

AD NAS 9-15200 Technical Memorandum Format Specifications for LACIE (Phase III) and Accuracy Assessment Computer Data Products.

ORIGINAL PAGE IS
OF POOR QUALITY

# 3.0 SYSTEM DESCRIPTION

## 3.1 HARDWARE DESCRIPTION

The software for CLASFYG is operational on the IBM 3031 computer at Purdue.

## 3.2 CLASFYG PROCESSOR SYSTEM FLOWCHART

The system level data flow diagram is shown in figure 3.1.

The following system flowchart assumes the use of the EXEC file described in Appendix C.

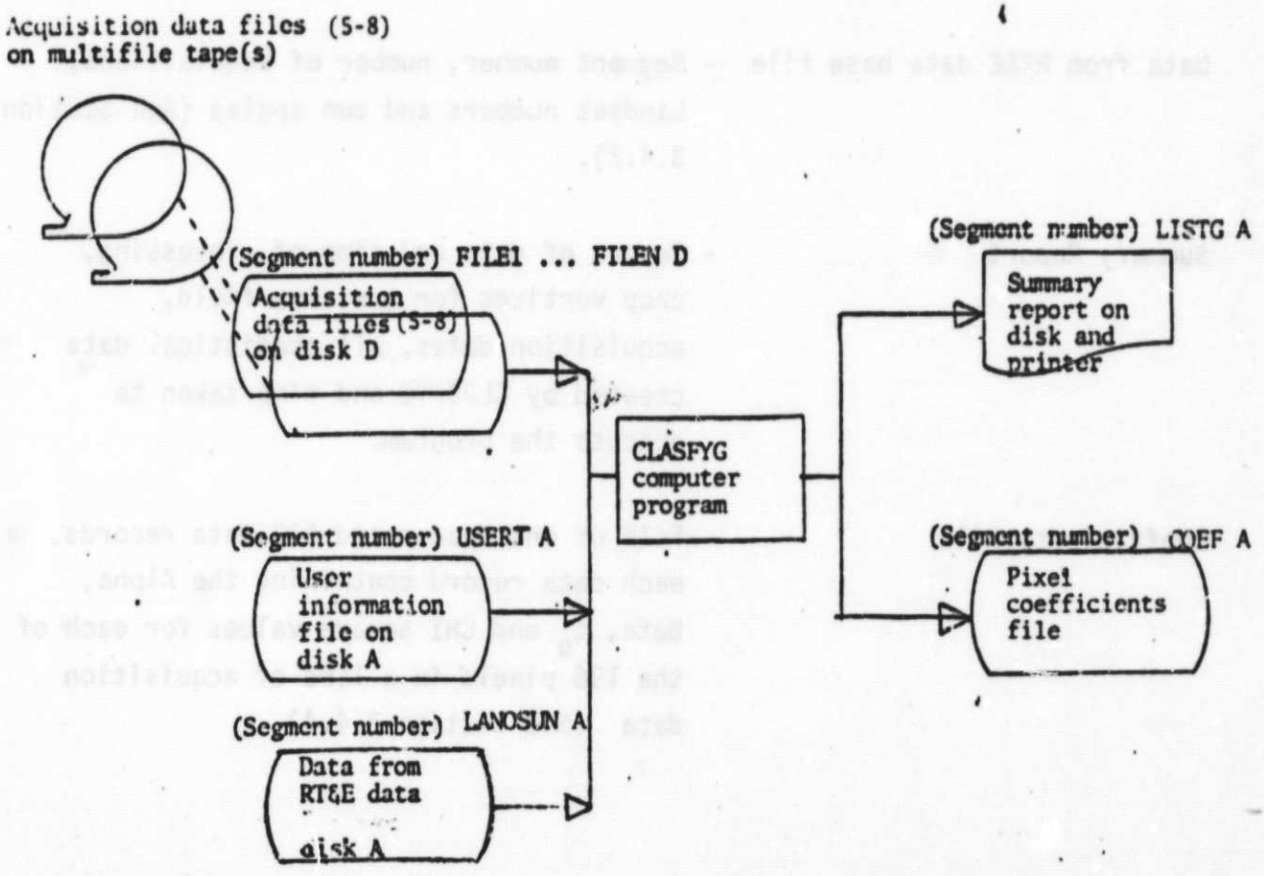



Figure 3.1

The inputs and outputs shown on Figure 3.1 are described briefly here. A more detailed description is given in section 3.4.

Acquisition data files - Header record and 4 channels of data for each of 5 to 8 acquisitions (See Section 3.4.3).

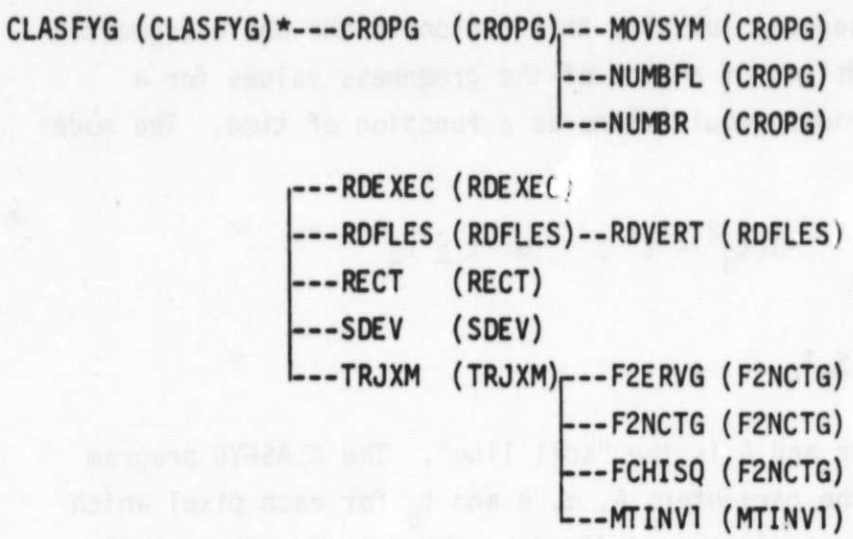
User information file - User information file supplying crop, initial guesses for greenness values, training field vertices and optional test field vertices (See Section 3.4.1).

Data from RT&E data base file - Segment number, number of acquisitions, Landsat numbers and sun angles (See Section 3.4.2).

Summary Report - Report of date and time of processing, crop vertices for training field, acquisition dates, all statistical data created by CLASFYG and time taken to execute the program.

Coefficients File - File of one header and 117 data records, each data record containing the Alpha, Beta, $t_0$ and CHI square values for each of the 196 pixels in a line of acquisition data (See Section 3.4.4).

CLASFYG Hierarchy Diagram




* Parentheses enclose the name of the file which contains the source statements for the named program which precedes them. For example,

PROGRAM (Source statement file name)

## 3.3 SOFTWARE DESCRIPTION

The CLASFYG program fits a model to the time trajectory of each pixel in a specified region of a segment (usually this region is the whole segment). The time trajectory consists of a plot of the greenness values for a particular pixel on various acquisitions as a function of time. The model is given by

$$G(t) = A \left(\frac{t}{t_o}\right)^{\alpha} e^{-\beta(t_o^2 - t^2)} \quad \text{for } t \geq t_o$$

and

$$G(t) = A \quad \text{for } t \leq t_o$$

where G is the greenness and A is the "soil line". The CLASFYG program produces estimates of the parameters A, $\alpha$, $\beta$ and $t_o$ for each pixel which give the best fit to the trajectory defined by the data for that pixel. It also calculates a value for $\chi^2$ which describes the goodness of fit.

The first stage of the processing consists of determining starting values for A, $\alpha$, $\beta$ and $t_o$ which are subsequently used as starting points for fitting the model to pixels in the test area.

These starting values are determined using the data in a training field selected by the user and a set of initial "guesses" for these parameters also input by the user. The training field should be a field of the crop of major interest. Also during this first stage, an array SGMX is calculated which contains the standard deviations of the pixels in the training field for each acquisition used. These are used in weighting the acquisitions when fitting the model to the pixels in the test area.

The second stage of the processing consists of fitting the model to each pixel in the test area starting the iteration procedure for each pixel with the starting values calculated in phase I. When this is completed the final values for $\alpha$, $\beta$, $t_o$ and $\chi^2$ for each pixel are output as a formatted disc file. Also a summary report is printed on the line printer.

## 3.4 FILE DESCRIPTION

### 3.4.1 USER INFORMATION FILE (Segment number) USERG A

The following records describe the name of the crop, the initial guesses for the coefficients for greenness and the training and test fields and allows the user to input any number of free-form comments.

In record types 1-5 the "keyword" is in columns 1-4 and any parameter or comment data is in columns 11 through 72. Numbers in a series are separated by commas; blanks are optional.

Record types 6 and 7 which describe the fields are blank in columns 1 through 10 and the field definition is in columns 11-72.

Record type 6 must follow the "*END" record and must preceed record type 7.

Record type 7 is optional, but if it exists it must follow record type 6.

1. "COMMENT" record
   EXAMPLE: COMMENT THIS IS AN EXAMPLE EXECUTION OF THE PROGRAM
   These records allow the user to enter any number of free-form comments.

2. "CROP" record
   EXAMPLE: CROP CORN
   This record specifies the 4 character name of the crop of interest. The name begins at the first non-blank character.

3. "GREENNESS" record
   EXAMPLE: GREENNESS 12.00, 11.50, 1.50, 1.50
   This record specifies the initial guesses for the A, Alpha, Beta and $t_0$ values of the crop in the training field.

4. "SUNANGLE" record

   EXAMPLE: SUNANGLE 45, 47, 51, 46, 42

   This record specifies the sun angles and is used only if the sun angles in the data base are wrong.

5. "*END" record

   This record specifies the end of the user information except for the definition of the training field and test field.

6. "TRAINING FIELD" record

   EXAMPLE: (1,1), (97,100), (97,120), (125,110), (100,100)

   This record specifies the coordinates of the corners of the training field. The format of the line is as follows:

   (1, 1) Dummy variable to be used for sample skip factor and line skip factor; not currently used.

   ($X_1$, $Y_1$) Upper leftmost corner of training field expressed as (sample, line).

   ($X_2$, $Y_2$) Upper rightmost corner of training field expressed as (sample, line).

   ($X_3$, $Y_3$) Lower rightmost corner of training field expressed as (sample, line).

   ($X_4$, $Y_4$) Lower leftmost corner of training field expressed as (sample, line).

7. "TEST FIELD" record

   EXAMPLE (1,1), (1,1), (196,1), (196,117), (1,117)

   This optional record specifies the coordinates of the corners of the test field. The default is the whole segment. The format is the same as the format for the training field record.

ORIGINAL PAGE IS
OF POOR QUALITY

.2 ACQUISITION DATA FILES

: following records of the 5 to 8 acquisitions data files are input to the
\SFYG program and are read by RDFLES.

ese files each contain 4 channels of LANDSAT data for one acquisition.
an acquisition is used twice the corresponding file will be duplicated.
te that these files are all in universal format. Record 1 is read with
17(180A1) format statement; records 2-118 are read with a 5(180A1) format
tatement.

| ecord | Bytes | Contents |
|---|---|---|
| | 1-3060 | Header record in universal format. |
| 2 | 1-72 | Filler. |
| | 73-268 | Gray level values for the 196 samples of and 1 for line 1. |
| | 269-464 | Gray level values for the 196 samples of band 2 for line 1. |
| | 465-660 | Gray level values for the 196 samples of band 3. for line 1. |
| | 661-856 | Gray level values for the 196 samples of band 4 for line 1. |
| 3 | 1-856 | Same data format as record 2; gray level values for bands 1-4 for line 2. |
| ⋮ | | |
| N | 1-856 | Same data format as record 2; gray level values for bands 1-4 for line N-1. |
| ⋮ | | |
| 118 | 1-856 | Same data format as record 2; gray level values for bands 1-4 for line 117. |

### 3.4.3 DATA FROM RT&E DATA BASE LANOSUN FILE D

The following records describe the segment number, the number of acquisitions, and the date, sun angle and Landsat numbers for each acquisition.

| Record | Bytes | Format | Contents |
|---|---|---|---|
| 1 | 1-4 | A4 | Segment number. |
| 2 | 1-4 | I4 | Number of acquisitions. |
| 3 | 1-56 | 8(2X,A5) | Acquisition dates. |
| 4 | 1-32 | 8I4 | Sun angles. |
| 5 | 1-32 | 8I4 | Landsat numbers. |

### 3.4.4 COEFFICIENTS FILE (Segment number) COEF A

The following records are output from the CLASFYG program:

Format

| Record | Bytes | Format | Contents |
|---|---|---|---|
| 1 | 1-3060 | 17(180A1) | Header record in Universal format. |
| 2-118 | 3136 | 18(45A4) | Coefficients Alpha, Beta, $t_o$ and CHI-square for the 196 pixels on 1 line. |

Detailed format

| Record | Words | Contents |
|---|---|---|
| 1 | 1-765 | Header record in Universal format. |
| 2 | 1 | Alpha value for pixel 1 in line 1 (computational integer). |
| 2 | 2 | Beta value for pixel 1 in line 1 (computational integer). |
| | 3 | $t_o$ value for pixel 1 in line i (computational integer). |
| | 4 | CHI-square value for pixel 1 in line 1 (computational integer). |
| | ⋮ | |
| | 1+N*4 | Alpha value for pixel N+1 in line 1 (computational integer). |
| | 2+N*4 | Beta value for pixel N+1 in line 1 (computational integer). |
| | 3+N*4 | $t_o$ value for pixel N+1 in line 1 (computational integer). |
| | 4+N*4 | CHI-square value for pixel N+1 in line 1 (computational integer). |
| | ⋮ | |
| 2 | 781 | Alpha value for pixel 196 in line 1 (computational integer). |
| | 782 | Beta value for pixel 196 in line 1 (computational integer). |
| | 783 | $t_o$ value for pixel 196 in line 1 (computational integer). |
| | 784 | CHI-square value for pixel 196 in line 1 (computational integer). |

Coefficients file (continued)

| Record | Words | Contents |
|---|---|---|
| 3 | 1-784 | Same data format as record 2; Alpha, Beta, $t_o$ and CHI-square values for 196 pixels of line 2. |
| : | | |
| M+1 | 1-784 | Same data format as record 2; Alpha, Beta, $t_o$ and CHI-square values for 196 pixels of line M. |
| : | | |
| 118 | 1-784 | Same data format as record 2; Alpha, Beta, $t_o$ and CHI-square values for 196 pixels of line 117. |

## 3.5 DETAILED SOFTWARE DESCRIPTION

### 3.5.1 CLASFYG

Purpose

CLASFYG uses a user determined training field in a full scene to determine trajectory coefficients for each band of radiance values to fit the model defined in section 1.0. These coefficients Alpha, Beta, $t_o$ and CHI-square are written for each pixel to the output file described in section 3.4.3.

Linkages

CLASFYG calls CROPG, RDEXEC, CPTIME, RECT, SDEV, TRJXM and RDFLES.

Interface

Calling sequence:

Not applicable (an EXEC that can be used to call CLASFYG is described in Section 4.0).

Calling sequence parameters:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

| Label | Variable | Element Position | Input/ Output | Description |
|---|---|---|---|---|
| /LINES/ | LINENO | 2 | Ø | Line number. |
| /LINES/ | IDATA(856,8) | 3 | I | Radiance values for a maximum of 8 channels. |
| /LINES/ | NOACQS | 4 | Ø | Number of acquisitions. |
| /MISC/ | NOACQ | 1 | I | Number of acquisitions. |
| /MISC/ | ACQDT(8) | 2 | I | Acquisition dates. |
| /MISC/ | KSEGM | 3 | I | Segment number. |
| /USER/ | ICROP(4) | 1 | I | Name of crop. |
| /USER/ | ZX(4) | 2 | I | Initial guesses for coefficients A, Alpha, Beta and $t_o$. |
| /USER/ | SANGGG(8) | 3 | Ø | Sun angle correction factor. |

Blank COMMON parameters:

None.

Inputs

| Unit | Type | Description |
|---|---|---|
| 7 | Seq. data | Data from RT&E data base (See Section 3.4.3). |
| 11 | Seq. data | Acquisition data (See Section 3.4.2). |
| 12 | Seq. data | Acquisition data (See Section 3.4.2). |
| 13 | Seq. data | Acquisition data (See Section 3.4.2). |
| 14 | Seq. data | Acquisition data (See Section 3.4.2). |
| 15 | Seq. data | Acquisition data (See Section 3.4.2). |
| 16 | Seq. data | Acquisition data (See Section 3.4.2) optional. |
| 17 | Seq. data | Acquisition data (See Section 3.4.2) optional. |
| 18 | Seq. data | Acquisition data (See Section 3.4.2) optional. |
| 21 | Seq. data | User defined parameters (See Section 3.4.1). |
| 30 | Seq. data | Scratch file to reformat data. |

Outputs

| Unit | Type | Description |
|---|---|---|
| 6 | Report | Summary report. |
| 8 | Seq. data | Header and coefficients Alpha, Beta, $t_o$ and CHI-square. |
| 30 | Seq. data | Scratch file to reformat data. |

Storage requirement

Not applicable.

Description

CLASFYG calls CROPG to read the user specified parameters crop of interest and initial guesses for greenness coefficients A, Alpha, Beta, $t_o$.

CLASFYG calls RDFLES to read the header record and the parameters for the training field.

CLASFYG then calls CPTIME to get the current time and date and writes the report heading.

CLASFYG calculates the minimum and maximum boundaries for pixels in the test field, calculates days elapsed from the beginning of the year as a multiple of 100 for each acquisition date and sets a switch if an acquisition date is used twice.

The rectangular field bounded by the minimum and maximum pixel numbers and line numbers is then processed using the following steps:

A line of radiance values is read.
Each pixel in the field is checked to be in the training field.
The values for the pixels in the training fields are used to calculate a greenness value which is saved in a matrix.

When sufficient pixels have been accumulated, the mean and standard deviation are calculated for the test field.

CLASFYG calls TRJXM to calculate improved values for A, Alpha, Beta and $t_o$ and a CHI-square value for the training field.

The header record is constructed from constants in the program, input from the user parameter file and the current date. The header record is written as the first record on the new classification file.

CLASFYG then evaluates the test field using the following steps:

A line of pixel data is read and the greenness value calculated TRJXM for every pixel on the line is called for each of the pixels.

The values for Alpha, Beta, $t_o$ and the CHI-square are calculated by TRJXM for each pixel.

The line of pixels coefficients for Alpha, Beta and $t_o$ and the CHI-square value is written to the coefficients file.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.2 SUBROUTINE CROPG

Purpose

CROPG reads and analyzes cards describing the initial guesses for A, alpha, beta and $t_0$ for the training field, the name of the crop and optionally the sun angle.

Linkages

CROPG is called by CLASFYG.

Interface

Calling sequence:

CALL CROPG (IERROR).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| IERROR | Ø | 0 if no error occurs;<br>1 if error occurs. |

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix.

| Label | Variable | Element Position | Input/ Output | Description |
|---|---|---|---|---|
| /USER/ | ICROP(4) | 1 | Ø | Name of crop of interest. |
| /USER/ | ZX(4) | 2 | Ø | Initial guesses for A, Alpha, Beta and $t_0$. |

Blank COMMON parameters:

None.

Inputs

| Unit | Type | Description |
|---|---|---|
| 21 | Seq. data | User information file (See Section 3.4.1) |

Outputs

None.

Storage requirement

Not applicable.

Description

Each record in the user defined parameter file is read and checked to have a valid control word in characters 1 through 4 defining the type of information in characters 11 through 80. If the control word is invalid, an error message is written and the error indicator is set nonzero; otherwise, the information is saved in the appropriate common block element.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.3 FUNCTION MOVSYM

Purpose

MOVSYM moves characters from array CARD to a series of left-justified words in array ITEMP.

Linkages

MOVSYM is called by CROPG.

Interface

Calling sequence:

```
KOUNT = MOVSYM (CARD, ITEMP).
```

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| CARD | I | 80 characters of user supplied data. |
| ITEMP | Ø | Array of left-justified words. |

Function Value:

| Name | Description |
|---|---|
| KOUNT | Count of symbols. |

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage Requirement

Not applicable.

Description

MOVSYM separates the characters in array CARD starting at character 11 and creates a series of left-justified words in array ITEMP.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.4 FUNCTION NUMBFL

**Purpose**

NUMBFL changes floating point numbers from characters in array CARD to computational numbers in arrary XNMVEC.

**Linkages**

NUMBFL is called by CROPG.

**Interface**

Calling sequence:

KOUNT = NUMBFL (CARD, XNMVEC).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| CARD | I | 80 characters of user supplied data. |
| XNMVEC | Ø | Array of floating point computational numbers. |

Function value:

| Name | Description |
|---|---|
| KOUNT | Count of floating point numbers. |

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

**Inputs**

None.

**Outputs**

None.

Storage requirement

Not applicable.

Description

NUMBFL separates the characters in array CARD starting at character 11 and creates a series of computational floating point numbers.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.5 RDEXEC

Purpose

RDEXEC reads the segment number, number of acquisitions, acquisition dates and Landsat numbers.

Linkages

RDEXEC is called by CLASFYG.

Interface

Calling sequence:

```
CALL RDEXEC.
```

Calling sequence parameters:

None.

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

| Label | Variable | Element Position | Input/ Output | Description |
|---|---|---|---|---|
| /USER/ | SUNANG(8) | 3 | Ø | Sun angle for each acquisition. |
| /MISC/ | NOACQ | 1 | Ø | Number of acquisitions. |
| /MISC/ | ACQDT(8) | 2 | Ø | Date for each acquisition. |
| /MISC/ | KSEGM | 3 | Ø | Segment number. |
| /MISC/ | LANDST(8) | 5 | Ø | Landsat number for each acquisition. |

Blank COMMON parameters:

None.

Inputs

| Unit | Type | Description |
|---|---|---|
| 7 | Seq. data | Data from RT&E data base (See Section 3.4.3). |

Outputs

None.

Storage requirement

Not applicable.

Description

RDEXEC reads the segment number, number of acquisitions, acquisition dates and Landsat numbers.

Flowchart

Not applicable.

Listing

See Appendix D for listing.

### 3.5.6 SUBROUTINE RDFLES

Purpose

RDFLES reads 1) the header record and one set of vertices or 2) one line of radiance values.

Linkages

RDFLES is called by CLASFYT.

Interface

Calling sequence:

CALL RDFLES (ITYPE, IDATA, X, Y, IERROR).

Calling sequence numbers:

| Argument | Input/ Output | Description |
|---|---|---|
| ITYPE | I | 1 to read header from each acquisition file and field definition; 2 to read 1 line of radiance values from each acquisition file. |
| IDATA(856,8) | Ø | Line of radiance values from each acquisition file. |
| X(4) | IØ | Array of sample vertices defining field. |
| Y(4) | IØ | Array of line vertices defining field. |
| IERROR | IØ | 0 - no error;<br>1 - error. |

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

| Label | Variable | Element Position | Input/ Output | Description |
|---|---|---|---|---|
| /LINES/ | NOACQS | 4 | I | Number of acquisitions. |
| /LINES/ | NXTLNE | 1 | IØ | Next line number. |

Blank COMMON parameters:

None.

Inputs

| Unit | Type | Description |
|---|---|---|
| 11 | Seq. data | Acquisition data for date 1 (See Section 4.3.2). |
| 12 | Seq. data | Acquisition data for date 2 (See Section 4.3.2). |
| 13 | Seq. data | Acquisition data for date 3 (See Section 4.3.2). |
| 14 | Seq. data | Acquisition data for date 4 (See Section 4.3.2). |
| 15 | Seq. data | Acquisition data for date 5 (See Section 4.3.2). |
| 16 | Seq. data | Acquisition data for date 6 (See Section 4.3.2). Optional. |
| 17 | Seq. data | Acquisition data for date 7 (See Section 4.3.2). Optional. |
| 18 | Seq. data | Acquisition data for date 8 (See Section 4.3.2). Optional. |

Outputs

None.

Storage requirement

Not applicable.

Description

If ITYPE=1, RDFLES reads the header record from each of the universal formatted acquisition data tapes and the vertices for one field. If ITYPE=2, RDFLES reads one line of radiance values for each acquisition.

Flowchart

Not applicable.

Listing

See Appendix D.

### 3.5.7 SUBROUTINE RDVERT

Purpose

RDVERT reads one set of vertices.

Linkages

RDVERT is called by CLASFYG.

Interface

Calling sequence:

CALL RDVERT (XFLD, YFLD, IERROR).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| XFLD(4) | Ø | Array of sample vertices defining field. |
| YFLD(4) | Ø | Array of line vertices defining field. |
| IERROR | Ø | 0 - no errors;<br>1 - error. |

Function value:

Not applicable.

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

| Unit | Type | Description |
|---|---|---|
| 21 | Seq. data | User specified data file (See Section 4.3.1). |

Outputs

None.

Storage requirement

Not applicable.

Description

RDVERT reads the vertices for one field.

Flowchart

Not applicable.

Listing

See Appendix D.

### 3.5.8 SUBROUTINE RECT

Purpose

RECT determines if a pixel lies within the specified vertices of a specific area.

Linkages

RECT is called by CLASFYT.

Interface

Calling sequence:

CALL RECT (LINE, PIXEL, YY, XX, ICOR).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| LINE | I | Current line number of pixel. |
| PIXEL | I | Current sample number of pixel. |
| YY(4) | I | Array of sample vertices defining field. |
| XX(4) | I | Array of line vertices defining field. |
| ICOR | Ø | 0 if pixel in field;<br>1 if pixel not in field. |

Function value:

Not applicable.

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

The pixel is checked to be a boundary pixel and considered in the training field if it is a boundary pixel. If the pixel is within the minimum and maximum limits of the specified area, the angles created between lines drawn between the pixel and boundary pixels are calculated and summed. If the sum of the angles is 360; the pixel is considered to be in the specified field.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.9 SUBROUTINE SDEV

Purpose

SDEV calculates the mean and standard deviation of a vector.

Linkages

SDEV is called by CLASFYT.

Interface

Calling sequence:

```
CALL SDEV (X, N, XB, S).
```

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| X(N) | I | Array of values. |
| N | I | Number of values in array X. |
| XB | Ø | Mean of values in array X. |
| S | Ø | Standard deviation of values in array X. |

Function value:

Not applicable.

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

SDEV calculates the mean (XB) and standard deviation (S) of vector X with N points.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.10 SUBROUTINE TRJXM

Purpose

TRJXM makes a least-square fit to a non-linear function to calculate coefficients for the curve.

Linkages

TRJXM is called by CLASFYG.

Interface

Calling sequence:

CALL TRJXM (X, Y, SIGMAY, NFILES, NTERMS, MODF, A DELTAA, SIGMAA, FLAMDA, YFIT, CHISQR).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| X | I | Array of Julian day/100 for acquisitions. |
| Y | I | Array of dependent variables. |
| SIGMAY | I | Array of errors on Y. |
| NFILES | I | Number of acquisitions. |
| NTERMS | I | Number of degrees of freedom. |
| MODE | I | Not used, was switch for calculation weight. |
| ESTIM | I,O | Array of coefficients A, Alpha, Beta and $t_o$. |
| DELTAA | I | Array of increments for A. |
| SIGMAA | O | Standard deviations. |
| FLAMDA | O | Proportion of gradient search. |
| YFIT | O | Array of fitted values. |
| CHISQR | o | Residual CHI square value. |

| Argument | Input/ Output | Description |
|---|---|---|
| IND | N/A | Not used. |
| WEIGHT | I | Array of weighting constants. |
| IPRINT | I | Print flag to use in debug. |

Function value:

Not applicable.

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

TRJXM calculates alpha and beta matrices, using the least-square method, for defining all parameters of the curve for the selected pixels in the training field.

The CHI-square value of the starting point is evaluated and the modified curvature matrix is inverted to find the new parameters.

The convergence depends on the CHI-square value and on the size of step.

If the CHI-square value increases, FLAMDA is increased and the process is re-initiated. If the CHI-square value descreases, the process is re-initiated until convergence is obtained. The parameters and uncertainties are evaluated.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.11 SUBROUTINE F2ERVG

**Purpose**

F2ERVG computes the derivatives of the function of each of the X values.

**Linkages**

F2ERVG is called by TRJXM.

**Interface**

Calling sequence:

CALL F2ERVG (X, I, ESTIM, DELTAA, NTERMS, DERIV).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| X | I | Array of time points for acquisition. |
| I | I | Current index in calling program. |
| ESTIM | IØ | Array of coefficients A, Alpha, Beta and $t_o$. |
| DELTAA | I | Array of increments for A. |
| NTERMS | I | Number of degrees of freedom. |
| DERIV | Ø | Array of derivatives with respect to parameters ESTIM. |

Function value:

Not applicable.

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

The radiance value and $t_o$ are set positive. Y is calculated as the radiance value/$t_o$. If Y is less than or equal to 1, the 4 derivatives are set to very small numbers; otherwise, the derivatives are set to calculated values.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.12 FUNCTION FCHISQ

Purpose

FCHISQ calculates the CHI-square value per degree of freedom.

Linkages

FCHISQ is called by TRJFIT and CHNFIT.

Interface

Calling sequence:

```
GOODFT=FCHISQ (Y, SIGMAY, NPTS, NFREE, MODE, YFIT, WEIGHT).
```

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| Y(NFILES) | I | Array of dependent variables. |
| SIGMAY(NFILES) | I | Array of errors on Y. |
| NPTS | I | Number of acquisitions. |
| NFREE | I | Number of degrees of freedom. |
| MODE | I | Not used. |
| YFIT(NFILES) | I | Array of fitted values. |
| WEIGHT(NFILES) | I | Array of weighting factors. |

Function value:

| Name | Description |
|---|---|
| GOODFT | CHI-square value per degree of freedom. |

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

FCHISQ calculates CHI-square value = $\xi( YFIT-Y)/SIGMA)^2$

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.13 FUNCTION F2NCTG

Purpose

F2NCTG calculates the value of DERIV from the radiance value, alpha and beta.

Linkages

F2ERVG is called by TRJXM.

Interface

Calling sequence:

CALL F2NCTG (X, I, ESTIM, IND).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| X(8) | I | Time array of Julian date/100 for each acquisition. |
| I | I | Index in calling program. |
| ESTIM(4) | I | Array of coefficients A, Alpha, Beta and $t_o$. |
| IND | N/A | Not used. |
| DELTAA(8) | I | Array incremental values. |
| NTERMS | I | Number of degrees of freedom. |
| DERIV(4) | Ø | Derivatives. |

Function value:

| Name | Description |
|---|---|
| YFIT(8) | Array of fitted values. |

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

F2NCTG calculates the value of the points on the curve from the values of alpha and beta and the current radiance value.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.14 SUBROUTINE MTINV1

Purpose

MTINV1 inverts a matrix of order NORDER.

Linkages

MTINV1 is called by TRJXM.

Interface

Calling sequence:

CALL MTINV1 (ARRAY, NORDER, DET).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| ARRAY(NORDER) | IØ | Matrix. |
| NORDER | I | Order of matrix. |
| DET | Ø | Determinate of matrix. |

Function value:

None.

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

MTINV1 uses the Gauss-Jordon reduction techniques.

Flowchart

Not applicable.

Listing

See Appendix D for program.

### 3.5.15 FUNCTION NUMBR

**Purpose**

NUMBR changes integers from character array CARD to computational numbers in array NUMVEC.

**Linkages**

NUMBR is called by CROPTM.

**Interface**

Calling sequence:

KOUNT = NUMBR (CARD, NUMVEC).

Calling sequence parameters:

| Argument | Input/ Output | Description |
|---|---|---|
| CARD | I | 80 characters of user supplied data in character format. |
| NUMVEC | Ø | Array of integer computational numbers. |

Function value:

| Name | Description |
|---|---|
| KOUNT | Number of computational integers |

Labeled COMMON parameters:

None.

Blank COMMON parameters:

None.

**Inputs**

None.

**Outputs**

None.

Storage requirement

Not applicable.

Description

NUMBR separates the characters in array card starting at the 11th character and creates a series of computational integers.

Flowchart

Not applicable.

Listing

See Appendix D for program.

# 4.0 OPERATION

## 4.1 OPERATING DESCRIPTION

CLASFYG is operational on the IBM 3031 computer at LARS, West Lafayette, Indiana.

The CLASFYG program is one of the programs of the BADHWAR SYSTEM.

CLASFYG requires the use of a D disk which is assigned as a temporary disk and an E disk which is used to load LARS routines onto. The user therefore, must not assign a disk to his machine using either MODE E or MODE D. These disks will be assigned as needed.

Prior to executing the CLASFYG program the user must establish on his A disk a USER INFORMATION file as described in Section 3.4.1.

## 4.2 COMMANDS DESCRIPTION

To execute CLASFYG the user will enter the following series of commands which invoke the JOB CONTROL SOFTWARE. These commands are divided into two classes namely (1) FUNCTION commands and (2) PROGRAM commands. The FUNCTION commands, which perform all the functions except executing the program, are reusable, i.e., once they are invoked they remain in effect until reentered. The PROGRAM commands, which simply execute the program, must be reentered each time the program is to be executed.

The following list gives the commands required to execute the CLASFYG program. They are all FUNCTION commands except the PROGRAM command CLASFYG. These commands are to be given in the listed order.

START

CLASFYG.....

END

The following sections describe each of the commands in detail. Input fields are separated by blanks. If more than one word is required to describe an input field, the description is enclosed in pointed brackets <>. If an input is optional the field is enclosed in square brackets []. Do not include these explanatory characters <> [] when actually submitting input to the computer. To enter a command the user types one input per defined input field and separates each field with a blank.

### 4.2.1 START

The START command spools the user's console file. The use of this command along with the END command will provide a listing of all information appearing on the user console file. (If running an interactive job this is the terminal.) If running a batch job this is a system defined device. The START command is invoked by the user typing the following:

START

### 4.2.2 CLASFYG

The CLASFYG command is a PROGRAM command is used to invoke the CLASFYG program. All landsat input data is from the LARS RT&E Data Base and a series of programs is required to perform the necessary interface. The following illustrates this software flow.

```
                        ..RTEER (LARS ROUTINE)
                        :
CLASFYG.....LCGINF.....:.SEGALL (LARS ROUTINE)
                        :
                        ..TSRTHS....IVALUE
```

For a detailed description of the above JOB CONTROL SOFTWARE see appendix C.

The CLASFYG command must not be given unless the user has established a user's file on his A disk under the file description <SEGMENT NUMBER> USERG as described in Section 3.4.1. Furthermore, the SEGMENT NUMBER must contain all 4 digits of the segment. For example: segment 0882 dictates that the USER INFORMATION file be named:

0882 USERG A

The CLASFYG command is invoked by the user typing the following.

CLASFYG  <ACQ.#1> <ACQ.#2> <ACQ.#3> <ACQ.#4> <ACQ.#5>...<ACQ.#8>

As described in Section 3.4.2, the CLASFYG program requires the user to specify a minimum of 5 segment acquisitions as input and a maximum of 8 may be specified.

The classification file output from an execution of CLASFYG is written to a file named <SEGMENT NUMBER> COEF A. The program inputs acquired from the LARS Data Base and the USER INFORMATION file is spooled to the HOUSTON printer.

### 4.2.3 END

This command closes the user's console file and causes a spooled copy to be sent to the HOUSTON printer. This command has no effect if the START command was not previously issued. The END command is invoked by the user typing the following:

```
END
```

## 4.3 OPERATING EXAMPLE

For our example we assume that the user has established on his A disk the required USER INFORMATION file. In addition since our example deals with an execution using segment 882 the USER INFORMATION file must be established under the filename 0882 and filetype USERG.

| COMMAND | EXPLANATION OR ACTION TAKEN |
|---|---|
| START | Spools the console file. |
| CLASFYG 0882 78150 78159 78186 78313 78221 78231 78267 | Executes the CLASFYG using 7 acquisitions of segment 882. |
| END | Closes the user's console file and spools the file tothe HOUSTON printer. |

# APPENDIX A

COMMON BLOCK /LINES/

| | |
|---|---|
| NXTLNE | Not currently used. |
| LINENO | Current line number. |
| IDATA(856,8) | Acquisition data. |
| NOACQS | Number of acquisitions. |

COMMON BLOCK /MISC/

| | |
|---|---|
| NOACQ | Number of acquisitions. |
| ACQDT(8) | Acquisition dates stored left justified in double precision word. |
| KSEGM | Segment number stored as 4 character word. |
| LANDST(8) | Landsat numbers. |

COMMON BLOCK /USER/

| | |
|---|---|
| ICROP(4) | Name of crop of interest. |
| ZX(4,10) | Initial guesses for A, Alpha, Beta and $t_o$ for greenness. |
| SANGGG(8) | Sun angles for acquisitions. |

# APPENDIX B

## CLASFYG Variable Definitions

| Variable | Definition |
|---|---|
| ABS | ABSOLUTE VALUE FUNCTION |
| ACQDT | ACQUISITION DATES IN ALPHANUMERIC CHARACTER |
| ACQDTB | ACQUISITION DATES IN 1 CHARACTER WORDS |
| ALPHBF | BUFFER FOR ALPHA VALUES |
| ANG | TEMP VALUE, USED IN SUN ANGLE CORRECTION |
| BETABF | BUFFER FOR BETA VALUES AFTER TEST PIXEL CALL TO TRJXM |
| BSEG | SEGMENT NUMBER IN CHARACTERS |
| CHIBUF | BUFFER FOR CH SQ. VALUES |
| CHISQ | FP CHI SQ. VALUE CALC. BY TRJXM FOR CURRENT TEST PIXEL |
| CHNVAL | HOLD AREA FOR CHANNEL RADIANCE VALUES |
| CLNDST | CONSTANT FOR CONVERTING LANDSAT 3 DATA |
| CPTIME | SUBROUTINE TO RETURN TODAY'S DATE AND TIME |
| DBUF | NOT USED |
| HDR | BUFFER FOR HEADER |
| IALPHA | 2ND APPRX VALUE, ALPHA, CALC BY TRJMX FOR TEST PIXEL |
| IBETA | 3RD APPRX VALUE, BETA, CALC BY TRJMX FOR CURRENT TEST PIXEL |
| ICHISQ | INTGR CHI SQ. VALUE CALC. FOR CURRENT TEST PIXEL BY TRJXM |
| ICOR | RETURN PARAMETER FROM RECT, 0=PIXEL IN FLD, 1=PIXEL NOT IN FLD |
| ICPTIM | INITIAL CP TIME IN HUNDREDTHS OF SECONDS |
| ICROP | CROP IN ALPHANUMERIC CHARACTERS |
| ICUT | NOT USED |
| IDATA | BUFFER AREA FOR LINES OF RADIANCE VALUES |
| IDATE | ACQUISITION DATES IN INTEGER FORMAT |
| IDAY | DAY IN CHARACTER FORMAT |
| IDIFFC | ELAPSED CP TIME IN HUNDREDTHS OF SECONDS |
| IDIFFV | ELAPSED VIRTUAL CP TIME IN HUNDREDTHS OF SECONDS |
| IDUPDT | INDEX TO DUPLICATE DATE, ZERO IF NO DUPLICATES |
| IHOLD1 | ARRAY OF 1 CHARACTER WORDS |
| IHOLD4 | ARRAY OF 4 CHARACTER WORDS |
| II | INDEX |
| ILIM | SET, BUT NOT USED |
| IMON | MONTH IN CHARACTER FORMAT |
| INCRE | TEMP VALUE USED IN UNPACKING PIXELS |
| INDEX | INDEX TO LANDSAT CORRECTION FACTOR |
| IPIXEL | CURRENT PIXEL |
| ITO | 4TH APPRX VALUE, TO, CALC FOR CURRENT TEST PIXEL BY TRJXM |
| ITYPE | INPUT TO SUBR. RDFLE TO READ HEADER(1) OR RADIANCE VALUES(2) |
| IVCPTM | INITIAL VIRTUAL CP TIME |
| IYEAR | 2 CHARACTER YEAR |

| VARIABLE | DEFINITION |
|---|---|
| JM | INDEX |
| JSEGM | SEGMENT NUMBER AS 1 CHARACTER WORDS |
| KBB | INDEX |
| KBLANK | CONSTANT OF BLANK |
| KEQUAL | CONSTANT OF EQUAL MARK |
| KORNER | INDEX TO VERTICES FOR TEST FIELD |
| KSEGM | SEGMENT NUMBER AS 4 CHARACTER WORDS |
| KURACQ | CURRENT ACQUISITION |
| KURCHN | CURRENT CHANNEL |
| KURPXL | CURRENT PIXEL |
| LA | INDEX |
| LANDST | ARRAY OF LANDSAT NUMBERS FOR ACQUISITIONS |
| LIMIT | MAX CALLS TO TRJXM FOR EACH PIXEL |
| LINENO | CURRENT LINE NUMBER |
| LMAX | MAXIMUM LINE NUMBER FOR BOTH FIELDS |
| LMIN | MINIMUM LINE NUMBER FOR BOTH FIELDS |
| LOC | TEMP VALUE USED IN UNPACKING RADIANCE VALUES |
| MAXPXL | MAXIMUM PIXEL |
| MINPXL | MINIMUM PIXEL |
| MODE | NOT USED, ORIGINALLY USED TO SELECT WEIGHTING METHOD |
| NCPTIM | ENDING CP TIME IN HUNDREDTHS OF SECONDS |
| NMODE | CONSTANT OF 0, USED IN CALL TO TRJXM |
| NOACQ | NUMBER OF ACQUISITIONS |
| NOACQS | NUMBER OF ACQUISITIONS |
| NOACQS | NUMBER OF ACQUISITIONS |
| NOPXLS | NUMBER OF PIXELS |
| NOTRPX | CURRENT NUMBER OF PIXELS IN TRAINING FIELD |
| NTERM | NUMBER OF DEGREES OF FREEDOM (4) |
| NTIME | CURRENT DATE, NOT USED |
| NVCPTM | ENDING VIRTUAL CP TIME IN HUNDREDTHS OF SECONDS |
| NXTLNE | NOT USED |
| PELTA | 8TH PARAMETER TRJXM, CAN BE USED IN NON-ANALYTIC DIFFERENTIATION |
| QELTA | NOT USED |
| QFTX | 11TH PARAMETER IN CALLING SEQUENCE TO TRJXM |
| RDFLES | SUBROUTINE TO READ FILE DATA |
| RECT | SUBROUTINE TO DETERMINE IF PIXEL IN USER DEFINED FIELD |
| SANG | SUN ANGLE ARRAY |
| SANGGG | USER INPUT SUN ANGLE |
| SDEV | SUBROUTINE TO CALCULATE MEAN AND ST. DEV. |
| SETUPG | SBUROUTINE TO READ USER INPUT DATA |
| SGMATR | ST. DEV. GREEN TRAIN FLD, INDEXED BY ACQ. NO. USED FOR TEST FLD |
| SIGMA | ERROR ON PARAMETERS FOR RETURN FROM TRJXM |
| SIGMAG | ST.DEV. FOR TRAINING FIELD |
| SIN | SINE FUNCTION |
| STIME | STARTING TIME OF DAY IN CHARACTERS FOR HEADER |

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

**VARIABLE DEFINITION**

| | |
|---|---|
| TIMECP | ELAPSED CP TIME IN MINUTES |
| TIMEVR | ELAPSED VIRTUAL CP TIME IN MINUTES |
| TODAY | TODAY'S DATE |
| TRAINA | 1ST APPRX VALUE CALC IN TRJXM FOR TRAIN FLD, USED IN TEST TRJXM |
| TRALPH | 2ND APPRX VALUE CALC. IN TRJXM FOR TRAIN FLD, USED IN TEST TRJXM |
| TRBETA | 3RD APPRX VALUE CALC IN TRJXM FOR TRAIN FLD, USED IN TEST TRJXM |
| TRJXM | SUBROUTINE TO CALCULATE FIT ON TRAINING FLD AND THEN TEST PIXEL |
| TRTO | 4TH APPRX VALUE CALC IN TRJXM FOR TRAIN FLD, USED IN TEST TRJXM |
| TZERO | 4TH APPRX VALUE, TO, CALC FOR CURRENT TEST PIXEL BY TRJXM |
| TOBUF | BUFFER FOR TO ARRAY AFTER TEST PIXEL CALLS TO TRJXM |
| VERTPX | USER SUPPLIED LINE VERTICES |
| WEIGHT | 1./ SIGMAY**2, WEIGHT FACTOR USED BY STATISTICAL ROUTINES |
| XBETA | FP 2ND APPRX VALUE CALC. BY TRJMX FOR CURRENT TEST PIXEL |
| XCHIN | INITIALLY 200, THEN CURRENT VALUE OF CHI SQ AFTER CALLS TO TRJXM |
| XDAY | FL. PT. VALUES FOR DAYS OF YEAR / 100. |
| XDIF | FL. PT. TEMP HOLD AREA |
| XGREEN | GREENNESS VALUES FOR TRAIN PIXELS |
| XISQR | CHI SQ. AFTER BOTHS CALLS TO TRJXM |
| XLAMDA | INCRE PARAMETER, SET 1.E-03 1ST CALL TO TRJXM, MODIFIED BY TRJXM |
| XLINE | FP USER SUPPLIED LINE NUMBER FOR CALL TO RECT |
| XMEAN | MEAN OF GREENNESS VALUES FOR TRAINING FIELD |
| XMNTRN | ARRAY OF MEANS FOR TRAINING FIELD ACQUISITIONS |
| XPIXEL | FP USER SUPPPLIED PIXEL NUMBER FOR CALL TO RECT |
| XPIXGR | ARRAY OF GREENNESS VALUES FOR PIXELS |
| XTH | TEMP |
| XTOL | TOLERANCE FACTOR USED FOR TRJXM FIT CHECK |
| VERTLN | USER SUPPLIED PIXEL VERTICES |
| YGREEN | GREENNESS VALUES FOR TRAINING FIELD |

# APPENDIX C

ACCEPTED OFF

CLASFYG EXEC
------------

HISTORY
-------

| | | | |
|---|---|---|---|
| CAROLINE HOUSTON | LARSCO | 11/01/80 | ORIGINAL EXEC |
| MARY A THOMPKINS | LARSCO | 01/29/81 | DEBUG |

PURPOSE
-------

THIS EXEC EXECUTES A FORTRAN PROGRAM (LCGINF) WHICH ACCESSES THE LARS USER DATA BASE FOR ACQUISITION REQUESTED. LISTING WRITES AN EXEC (CCFILES) WHICH CREATES DATA FILES FOR EACH ACQ. AFTER EXECUTION OF THE ABOVE THE CLASFYG PROGRAM IS LOADED.

ARGUMENTS TO THE EXEC ARE AS FOLLOWS:

| ARGUMENT NO. | DESCRIPTION |
|---|---|
| &1 | SEGMENT NUMBER |
| &2 | ACQ. NUMBER |
| &3 | ACQ. NUMBER |
| &4 | ACQ. NUMBER |
| &5 | ACQ. NUMBER |
| &6 | ACQ. NUMBER |
| &7 | ACQ. NUMBER |
| &8 | ACQ. NUMBER |
| &9 | ACQ. NUMBER |

*ARGUMENTS: 5 ACQ. NUMBERS ARE REQUIRED. THE PROGRAM WILL ACCEPT 9.

FILE DEFINITION/DESCRIPTION FOR ALL FILES USED IN THESE PROGRAMS AND EXECS ARE AS FOLLOWS:

| UNIT | DESCRIPTION |
|---|---|
| 2 | LISTING EXEC |
| 3 | TERMINAL READ |
| 4 | LARS SCALL ROUTINE |
| 5 | TERMINAL WRITE LARS ERROR MESSAGES |
| 6 | PRINTER FILE STORED IN FILE A1 LISTG |
| 7 | INPUT FILE-LANDSAT NUMBER EACH ACQ |
| 8 | COEFFICIENTS FILE STORED IN A1 COEF |
| 9 | USED IN HARDWAR SYSTEM GROUND TRUTH |
| 10 | USED IN HARDWAR SYSTEM CLASS/CLUSTER |
| 11 | A1 FILE1 0 -- RADIANCE FILE1 |
| 12 | A1 FILE2 0 -- RADIANCE FILE2 |
| 13 | A1 FILE3 0 -- RADIANCE FILE3 |
| 14 | A1 FILE4 0 -- RADIANCE FILE4 |
| 15 | A1 FILE5 0 -- RADIANCE FILE5 |
| 16 | A1 FILE6 0 -- RADIANCE FILE6 |
| 17 | A1 FILE7 0 -- RADIANCE FILE7 |
| 18 | A1 FILE8 0 -- RADIANCE FILE8 |
| 19 | HARDWAR SYSTEM |
| 20 | PRIMARY STATION -- STORED IN CODOC FILE A |
| 21 | A1 USERG -- USER CONTROL PARAMETER FILE |
| 22 | GT FILE RECALL FILE |
| 23 | CLASS FILE RECALL FILE |
| 24-29 | HARDWAR SYSTEM |
| 30 | RESERVED UNIT |

EXCEPTIONS
----------

THE FOLLOWING ERROR CONDITIONS CAUSE PROGRAM TERMINATION:

1. NO TERMINAL OR DISK AVAILABLE.
2. INCORRECT INPUT PARAMETERS INPUT TO PROGRAM.
3. ERRORS IN ACCESSING LARS DATA BASE.

PROCEDURE
---------

ASSIGN A [illegible] SPECIFY FILE [illegible] ASSIGN [illegible]



ORIGINAL PAGE IS
OF POOR QUALITY

```
*
      &SPACE 1
      &TYPE CLASFYG &1 &2 &3 &4 &5 &6 &7 &8 &9
      TAG DEV PRINTER HOUSTON
      SPOOL PRINTER CONT NOHOLD TO RSCS
      GLOBAL TXTLIB CMSLIB FORTMOD2
      CP QUERY VIRTUAL 192
      &IF &RETCODE NE 0 GETDISK TEMP 24 CLEAR
      &IF &RETCODE NE 0 &TYPE NO TEMP DISK ACCESSED.
      &IF &RETCODE NE 0 &EXIT 1
*
*
* GET LARS DISK WHICH CONTAINS RISE DATA BASE
      GETDISK RSCDISK 19A F
*
*
* FILEDEF TERMINAL AND FILEDEF EXEC FILE WRITTEN DURING RUN.
*
      FILEDEF 5 TERM (PERM
      FILEDEF 6 TERM (PERM
      FILEDEF FT02F001 DISK LCGINFO EXEC D(LRECL 80 BLKSIZE 80 PERM
      FILEDEF FT07F001 DISK LARDSUM FILE D (LRECL 80 BLKSIZE 80 PERM
      FILEDEF FT20F001 DISK CGDOC FILE A (LRECL 80 BLKSIZE 80 PERM
*
*
* CHECK FOR ACCEPTABLE PARAMETER COUNT
*
      &IF &INDEX LT 4 &GOTO -NO
      &IF &INDEX LE 9 &GOTO -TRUE
-NO   &TYPE TOO FEW-TOO MANY INPUTS
      &EXIT 2
*
-TRUE      &ACQCNT = &INDEX - 1
*
*
* IF LENGTH OF &1 (SEGMENT NUMBER) < 4 CONCATENATE 0.
*
-LOP  &SEGLNG = &LENGTH &1
      &IF &SEGLNG EQ 4 &SKIP 3
      &1 = &CONCAT 0 &1
      &SEGLNG = &SEGLNG + 1
      &GOTO -LOP
*
*
* STACK ALL EXEC INPUT TO ALLOW LCGINF TO ACQUIRE TAPE NUMBERS
* FILE NUMBERS,SATELLITE NUMBERS,AND SUN ANGLES FOR EACH ACQ.
*
      &STACK &ACQCNT
      &COUNT = 9
      &LOOP 2 &INDEX
      &COUNT = &COUNT + 1
      &STACK &&COUNT
*
*
* LOAD FORTRAN PROGRAM WHICH WILL ACCESS LARS RISE DATA BASE.
* PROGRAM WILL WRITE LCGINFO EXEC D. FILES FOR EACH ACQ.WILL BE
* TRANSFERRED FROM TAPE TO DISK IN LCGINFO EXEC.FILE DEF FOR FILES
* WILL BE DONE.
*
      LOAD LCGINF SEGALLOC RISERDB (NOMAP CLEAR START
      REL 19A
      EXEC LCGINFO D
      &READ VARS &ERR
      &IF &ERR EQ 1 &EXIT 3
*
*
* FILE DEF FOR USER DEFINED CONTROL FILE AND OUTPUT FILE.
*
      FILEDEF FT06F001 DISK &1 LISTG (LRECL 132 BLKSIZE 132 PERM
      FILEDEF 9 DISK &1 COEF (LRECL 3168 BLKSIZE 3168 RECFM F PERM
      FILEDEF FT21F001 DISK &1 USERIG &1 (BLKSIZE 80 PERM
*
*
      LOAD CLASFYG (CLEAR START NOMAP
*
*
* PRINT REPORT
*
      PRINT CGDOC FILE A
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: CLASEYG  EXEC     A   (AUS / PURDUE UNIVERSITY
      PRINT *) (LIST
      SPOOL PUNCH* CLOSE
      RETT
&END
```



ORIGINAL PAGE IS
OF POOR QUALITY

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

```
                C     PROGRAM LCGINF                                                        LCG00010
                CCCCC                                                                       LCG00020
                CCCCC                    GET INFO FROM LARS/WRITE EXEC FOR CLSFYG          LCG00030
                CCCCC ------------------------------------------------------------------   LCG00040
                CCCCC                                                                       LCG00050
                CCCCC HISTORY                                                               LCG00060
                CCCCC -------                                                               LCG00070
                CCCCC                                                                       LCG00080
                CCCCC     M A TOMPKINS      LEMSCO      01/27/81      ORIGINAL CODE         LCG00090
                CCCCC                                                                       LCG00100
                CCCCC METHOD                                                                LCG00110
                CCCCC ------                                                                LCG00120
                CCCCC                                                                       LCG00130
                CCCCC     READ NUMBER OF ACQS.,SEGMENT NUMBER,AND ACQ. NUMBERS.ACCESS THE   LCG00140
                CCCCC     LARS RT&E DATA BASE. IF SUCESSFUL WRITE LSTINFO EXEC TO MOUNT     LCG00150
                CCCCC     TAPES AND TRANSFER DATA TO DISC FILES. FILEDEF ALL NEW DISK FILES.LCG00160
                CCCCC     WRITE INPUT FILE CONSISTING OF NUMBER OF ACQ,SATELITE NUMBER      LCG00170
                CCCCC     AND SUN ANGLES FOR EACH ACQUISITION. WRITE SEGMENT NUMBER AND     LCG00180
                CCCCC     ACQUISITION NUMBERS FOR DOCUMENTARY PURPOSES.                     LCG00190
                CCCCC                                                                       LCG00200
                CCCCC MACHINE DEPENDENT CODE                                                LCG00210
                CCCCC ----------------------                                                LCG00220
                CCCCC                                                                       LCG00230
                CCCCC     NONE.                                                             LCG00240
                CCCCC                                                                       LCG00250
                CCCCC EXTERNAL REFERENCES                                                   LCG00260
                CCCCC -------------------                                                   LCG00270
                CCCCC                                                                       LCG00280
                CCCCC     SEGALL    LARS ROUTINE TO ACQUIRE INFO FROM RT&E DATA BASE       LCG00290
                CCCCC     TSRTHS    TAGSORT USING HIBBARD'S SHELL SORT                     LCG00300
                CCCCC     RTEERR    LARS ERROR ROUTINE                                     LCG00310
                CCCCC                                                                       LCG00320
                CCCCC EXCEPTIONS                                                            LCG00330
                CCCCC ----------                                                            LCG00340
                CCCCC                                                                       LCG00350
                CCCCC     IF IERR <> 0 OR 4 WRITE ERROR MESSAGE AND WRITE EXEC              LCG00360
                CCCCC     TO TERMINATE PROGRAM.                                             LCG00370
                CCCCC                                                                       LCG00380
                CCCCC LOCAL DECLARATIONS                                                    LCG00390
                CCCCC ------------------                                                    LCG00400
                C                                                                           LCG00410
ISN 0002        C     INTEGER INDEX(89,112)                                                 LCG00420
                C                                  INFO ON EACH ACQ REQUESTED FORM DATA BASE LCG00430
ISN 0003        C     INTEGER LSEGNO                                                        LCG00440
                C                                  LANDSAT SEGMENT NUMBER                   LCG00450
ISN 0004        C     INTEGER NACQRQ                                                        LCG00460
                C                                  NUMBER OF ACQUISITION REQUESTED          LCG00470
ISN 0005        C     INTEGER LSEGTP(64)                                                    LCG00480
                C                                  TEMP FOR TESTING OF LSEGAQ               LCG00490
ISN 0006        C     INTEGER NUMACQ                                                        LCG00500
                C                                  NO. OF ACQ REQUESTED: NO. OF ACQ. FOUND. LCG00510
ISN 0007        C     INTEGER LSEGAQ(64)                                                    LCG00520
                C                                  LANDSAT SEG. ACQ. DATE                   LCG00530
ISN 0008        C     INTEGER IERR                                                          LCG00540
                C                                  RT&E DATA BASE ERROR CODE                LCG00550
ISN 0009        C     INTEGER LACQRQ(5,10)                                                  LCG00560
                C                                  LANDSAT INFO REQUESTED BY USER           LCG00570
ISN 0010        C     INTEGER NACQRQ                                                        LCG00580
                C                                  NUMBER OF ACQ. REQUESTED                 LCG00590
ISN 0011        C     INTEGER INTAG(10)                                                     LCG00600
                C                                  ORDER TAGS (INDEX)                       LCG00610
ISN 0012        C     INTEGER LSTAPE                                                        LCG00620
                C                                  LAST TAPE NUMBER                         LCG00630
ISN 0013        C     INTEGER NFLSKP                                                        LCG00640
                C                                  NUMBER OF FILES TO SKIP                  LCG00650
ISN 0014        C     INTEGER IUNIT                                                         LCG00660
                C                                  UNIT NUMBER FOR FILE DEFINITION          LCG00670
                C                                                                           LCG00680
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
                C
                C PROCEDURE
                C ---------
                C
                C
                C READ FROM STACK THE NUMBER OF ACQ NEEDED,SEGMENT NUMBER, AND
                C SPECIFIC ARQUISITIONS SOUGHT.
                C
                C
                C START WRITING LSTINFO EXEC.
                C
ISN 0015                WRITE(2,90)
ISN 0016             90 FORMAT(' &CONTROL OFF')
                C
ISN 0017                READ(3,100)NUMACQ
ISN 0018            100 FORMAT(I1)
                C
ISN 0019                READ(3,110)LSEGNO
ISN 0020            110 FORMAT(I4)
                C
ISN 0021                DO 105 I = 1,NUMACQ
ISN 0022                   READ(3,120)LSEGAQ(I)
ISN 0023            105 CONTINUE
                C
ISN 0024            120 FORMAT(I5)
                C
                C
                C WHEN CALLING LARS SEGALL ROUTINE THE ARRAY OF ACQ MUST NOT
                C CONTAIN ANY DUPLICATE ACQ NUMBERS. THE LSEGTP ARRAY CONSIST
                C OF UNIQUE ACQ NUMBERS.NACQRQ IS THE COUNT OF UNIQUE NUMBERS.
                C
ISN 0025                NACQRQ = 0
ISN 0026                DO 140 K = 1,NUMACQ
ISN 0027                   DO 130 J = 1,NUMACQ
ISN 0028                      IF(LSEGAQ(K).EQ.LSEGTP(J)) GO TO 140
ISN 0030            130    CONTINUE
ISN 0031                   NACQRQ = NACQRQ + 1
ISN 0032                   LSEGTP(NACQRQ) = LSEGAQ(K)
ISN 0033            140 CONTINUE
                C
                C
                C CALL LARS SEGALL ROUTINE FOR ALL INFO ON THE REQUESTED ACQ.
                C BECAUSE OF THE POSSIBILITY OF DUPLICATE INFO ON SOME SEGMENTS
                C AND THE POSSIBILITY THAT THE ORDER OF THE INFO RETURNED MAY BE
                C DIFFERENT FROM REQUESTED SEG ORDER A SEARCH IS REQUIRED TO ORDER
                C AND DELETE IF ANY DUPLICATE INFO.
                C
ISN 0034                CALL SEGALL(LSEGNO,NACQRQ,LSEGTP,INDEX,IERR,4,'E')
                C
                C
                C CHECK IERR MESSAGE IF <> 0 OR 4 CALL LARS RTEERR ROUTINE FOR
                C ERROR MESSAGE.
                C
ISN 0035                IF(IERR.EQ.0.OR.IERR.EQ.4) GO TO 300
ISN 0037                CALL RTEERR(IERR,5)
ISN 0038                WRITE(2,250)
ISN 0039            250 FORMAT(' &STACK 1',/,' &EXIT')
ISN 0040                GO TO 900
                C
ISN 0041            300 WRITE(2,310)
ISN 0042            310 FORMAT(' &STACK 0')
                C
                C
                C SEARCH INFO RETURNED FROM DATA BASE. STORE INFO INTO LACQRQ
                C ARRAY .
                C
ISN 0043                DO 450 J = 1,NUMACQ
ISN 0044                   DO 370 I = 1,NACQRQ
ISN 0045                      ITEM = I
ISN 0046                      IF(INDEX(12,I).EQ.LSEGAQ(J)) GO TO 400
ISN 0048            370    CONTINUE
ISN 0049            400    LACQRQ(1,J) = INDEX(1,ITEM)
ISN 0050                   LACQRQ(2,J) = INDEX(2,ITEM)
ISN 0051                   LACQRQ(3,J) = INDEX(43,ITEM)
ISN 0052                   LACQRQ(4,J) = INDEX(57,ITEM)
ISN 0053                   LACQRQ(5,J) = INDEX(1,ITEM)*1000 + INDEX(2,ITEM)
ISN 0054            450 CONTINUE
                C
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
                C                                                                       LCG01520
                CC SORT INTAG ACCORDING TO SORT KEY MADE BY CONCATNATING TAPE#FILE#      LCG01530
                C                                                                       LCG01540
ISN 0055            CALL TSRTHS(INTAG, LACQRQ,5,NUMACQ,5,'A')                           LCG01550
                C                                                                       LCG01560
                CC                                                                      LCG01570
                CC WRITE PORTION OF EXEC TO TRANSFER TAPE FILES TO DATA FILES           LCG01580
                CC IF AN ACQ IS REUSED THE DISK FILE IS DUP.                            LCG01590
                C                                                                       LCG01600
ISN 0056            LSTAPE = 0                                                          LCG01610
ISN 0057            DO 600 I = 1,NUMACQ                                                 LCG01620
ISN 0058              IF(LSTAPE.EQ.LACQRQ(1,INTAG(I)))GO TO 520                         LCG01630
                C                                                                       LCG01640
                CC                                                                      LCG01650
                CC NO TAPE IS MOUNTED OR A NEW ONE IS TO BE MOUNTED                     LCG01660
                C                                                                       LCG01670
ISN 0060            WRITE(2,500)LACQRQ(1,INTAG(I))                                      LCG01680
ISN 0061      500   FORMAT(' TAPMOUNT ',I4,' TAP1 RG 1600')                           LCG01690
ISN 0062            WRITE(2,510)                                                        LCG01700
ISN 0063      510   FORMAT(' TAPE REW(TAP1')                                          LCG01710
ISN 0064            NFLSKP = LACQRQ(2,INTAG(I))-1                                       LCG01720
                C                                                                       LCG01730
                CC                                                                      LCG01740
                CC TAPE IS MOUNTED CONTINUE WRITING EXEC.                               LCG01750
                C                                                                       LCG01760
ISN 0065      520   IF(NFLSKP.GT.0)WRITE(2,530)NFLSKP                                 LCG01770
ISN 0067      530   FORMAT(' TAPE FSF ',I4)                                           LCG01780
ISN 0068            IF(NFLSKP.GE.0)WRITE(2,540)                                         LCG01790
ISN 0070      540   FORMAT(' FILEDEF INMOVE TAP1 (BLOCK 3060 RECFM U PERM')           LCG01800
ISN 0071            IF(NFLSKP.LT.0)WRITE(2,550)LSEGNO,INTAG(I-1)                        LCG01810
ISN 0073      550   FORMAT(' FILEDEF INMOVE DISK ',I5,' FILE',I1,                     LCG01820
                   &          ' D(LRECL 3060 BLOCK 3060 RECFM U PERM')                LCG01830
ISN 0074            WRITE(2,560)LSEGNO,INTAG(I)                                         LCG01840
ISN 0075      560   FORMAT(' FILEDEF OUTMOVE DISK ',I5,' FILE',I1,                    LCG01850
                   &          ' D(LRECL 3060 BLOCK 3060 RECFM U PERM')                LCG01860
ISN 0076            WRITE(2,580)                                                        LCG01870
ISN 0077      580   FORMAT(' MOVEFILE')                                               LCG01880
ISN 0078            LSTAPE = LACQRQ(1,INTAG(I))                                         LCG01890
ISN 0079            NFLSKP = LACQRQ(2,INTAG(I+1))-(LACQRQ(2,INTAG(I))+1)              LCG01900
ISN 0080      600 CONTINUE                                                              LCG01910
                C                                                                       LCG01920
                CC                                                                      LCG01930
                CC WRITE FILEDEF FOR THE ABOVE FILES                                    LCG01940
                C                                                                       LCG01950
ISN 0081            DO 620 I = 1,NUMACQ                                                 LCG01960
ISN 0082              IUNIT = INTAG(I) + 10                                             LCG01970
ISN 0083              WRITE(2,610)IUNIT,LSEGNO,INTAG(I)                                 LCG01980
ISN 0084      610     FORMAT(' FILEDEF FT',I2.'F001 DISK ',I5,' FILE',                  LCG01990
                   &          I1,' D(LRECL 3060 BLOCK 3060 RECFM U PERM')             LCG02000
ISN 0085      620 CONTINUE                                                              LCG02010
                C                                                                       LCG02020
                CC                                                                      LCG02030
                CC DETACH TAPE DRIVE                                                    LCG02040
                C                                                                       LCG02050
ISN 0086            WRITE(2,630)                                                        LCG02060
ISN 0087      630 FORMAT(' DETACH 181')                                                 LCG02070
                C                                                                       LCG02080
                CC                                                                      LCG02090
                CC WRITE NO. OF ACQ. SUN ELEVATION AND SATELITE NUMBER TO PROGRAM INPUT LCG02100
                CC FILE.                                                                LCG02110
                C                                                                       LCG02120
ISN 0088            WRITE(7,635)LSEGNO                                                  LCG02130
ISN 0089      635   FORMAT(I4)                                                        LCG02140
ISN 0090            WRITE(7,640)NUMACQ                                                  LCG02150
ISN 0091      640   FORMAT(I4)                                                        LCG02160
ISN 0092            WRITE(7,650)(LSEGAQ(I),I = 1,NUMACQ)                                LCG02170
ISN 0093      650   FORMAT(8I7)                                                       LCG02180
ISN 0094            WRITE(7,700)(LACQRQ(3,I),I = 1,NUMACQ)                              LCG02190
ISN 0095            WRITE(7,700)(LACQRQ(4,I),I = 1,NUMACQ)                              LCG02200
ISN 0096      700   FORMAT(10I4)                                                      LCG02210
                C                                                                       LCG02220
                CC                                                                      LCG02230
                CC FOR DOCUMENTARY PURPOSES WRITE SEGMENT NUMBER AND ACQUISITION        LCG02240
                CC NUMBER REQUESTED BY USER.                                            LCG02250
                C                                                                       LCG02260
ISN 0097            WRITE(20,750)                                                       LCG02270
ISN 0098      750   FORMAT(' PROGRAM INPUTS:')                                        LCG02280
ISN 0099            WRITE(20,800)LSEGNO                                                 LCG02290
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)          MAIN          OS/360  FORTRAN H EXTENDED          DATE 81.141/13.54.09      PAGE   4

   ISN 0100        800 FORMAT(' SEGMENT NUMBER:',I5)                                    LCG02300
   ISN 0101            WRITE(20,810)(LSEGAQ(I),I = 1,NUMACQ)                            LCG02310
   ISN 0102        810 FORMAT(' ACQUISITIONS:',8I10)                                    LCG02320
   ISN 0103            WRITE(20,820)(LACQRQ(3,I),I = 1,NUMACQ)                          LCG02330
   ISN 0104        820 FORMAT(' SUN ANGLES',10I4)                                       LCG02340
   ISN 0105            WRITE(20,830)(LACQRQ(4,I),I = 1,NUMACQ)                          LCG02350
   ISN 0106        830 FORMAT(' LANDSAT NUMBER',10I4)                                   LCG02360
   ISN 0107        900 STOP                                                             LCG02370
   ISN 0108            END                                                              LCG02380


                        *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0021  0022  0044  0045  0046  0057  0058  0060  0064  0071  0074  0078  0079  0079  0081  0082  0083  0092  0092
         0092  0094  0094  0095  0095  0095  0101  0101  0103  0103  0103  0105  0105  0105
J        0027  0028  0043  0046  0049  0050  0051  0052  0053
K        0026  0028  0032
O        0068
IERR     000A  0034  0035  0035  0037
ITEM     0045  0049  0050  0051  0052  0053  0053
INDEX    0002  0034  0046  0049  0050  0051  0052  0053  0053
INTAG    0011  0055  0058  0060  0064  0071  0074  0078  0079  0079  0082  0083
IUNIT    0014  0082  0083
LACQRQ   0009  0049  0050  0051  0052  0053  0055  0058  0060  0064  0078  0079  0079  0094  0095  0103  0105
LSEGAQ   0007  0022  0028  0032  0046  0092  0101
LSEGNO   0003  0019  0034  0071  0074  0083  0088  0099
LSEGTP   0005  0028  0032  0034
LSTAPE   0012  0056  0058  0078
NACQRQ   0004  0010  0025  0031  0031  0032  0034  0044
NFLSKP   0013  0064  0065  0065  0068  0071  0079
NUMACQ   0006  0017  0021  0026  0027  0043  0055  0057  0081  0090  0092  0094  0095  0101  0103  0105
RTEERR   0037
SFGALL   0034
TSRTHS   0055


                        *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL   DEFINED   REFERENCES
    90     0016    0015
   100     0018    0017
   105     0023    0021
   110     0020    0019
   120     0024    0022
   130     0030    0027
   140     0033    0026  0028
   250     0039    0038
   300     0041    0035
   310     0042    0041
   370     0048    0044
   400     0049    0046
   450     0054    0043
   500     0061    0060
   510     0063    0062
   520     0065    0058
   530     0067    0065
   540     0070    0068
   550     0073    0071
   560     0075    0074
   580     0077    0076
   600     0080    0057
   610     0084    0083
   620     0085    0081
   630     0087    0086
   635     0089    0088
   640     0091    0090
   650     0093    0092
   700     0096    0094  0095
   750     009A    0097
   800     0100    0099
   810     0102    0101
   820     0104    0103
   830     0106    0105
   900     0107    0040

                                        /    MAIN /    SIZE OF PROGRAM 009854 HEXADECIMAL BYTES

      NAME    TAG   TYPE  ADD.        NAME    TAG   TYPE  ADD.        NAME    TAG   TYPE  ADD.        NAME    TAG   TYPE  ADD.
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
      I SF    |*4  000344      J SF    |*4  000348      K SF    |*4  00034C      O       R*4  000350
   IERR SFA   |*4  000354   ITEM SF    |*4  000358   INDEX SFA  |*4  000664   INTAG SFA  |*4  000374
  IUNIT SF    |*4  000360 IBCOM# F  XF  |*4  000000  LACORG SFA  |*4  00039C  LSEGAG SF   |*4  000464
 LSEGNO SFA   |*4  000368 LSEGTP SFA    |*4  000564  LSTAPE S    |*4  000364  NACORG SFA  |*4  000368
 NFLSKP SF    |*4  000360 NUMACO SFA    |*4  000370  RTEERR SF XF|    000000  SEGALL SF XF|    000000
 TSRTHS SF XF      000000
```

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 105 | 23 | 0092FA | 130 | 30 | 009334 | 140 | 33 | 009360 | 300 | 41 | 009388 |
| 370 | 48 | 0093F0 | 400 | 49 | 0093FC | 450 | 54 | 00947C | 520 | 65 | 009512 |
| 600 | 80 | 009624 | 620 | 85 | 00967A | 900 | 107 | 009824 | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100000 | 1 | 009280 | 100001 | 22 | 0092D8 | 100002 | 24 | 009306 | 100003 | 27 | 009316 |
| 100004 | 28 | 00931A | 100005 | 31 | 009340 | 100006 | 34 | 009374 | 200001 | 36 | 009388 |
| 100007 | 37 | 009394 | 100008 | 44 | 0093D0 | 100009 | 45 | 0093D4 | 100011 | 55 | 009488 |
| 100012 | 58 | 00949E | 100013 | 60 | 009488 | 100014 | 66 | 00951C | 100015 | 68 | 009538 |
| 100016 | 69 | 009554 | 100017 | 71 | 009568 | 100018 | 72 | 009572 | 100019 | 74 | 00959E |
| 100020 | 81 | 009630 | 100021 | 82 | 009634 | 100022 | 86 | 009686 | 200002 | 103 | 0097C4 |

FORMAT STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 90 | 16 | 000028 | 100 | 18 | 000039 | 110 | 20 | 000030 | 120 | 24 | 000041 |
| 250 | 39 | 000045 | 310 | 42 | 000058 | 500 | 61 | 000068 | 510 | 63 | 000087 |
| 530 | 67 | 000099 | 540 | 70 | 0000A9 | 550 | 73 | 0000DA | 560 | 75 | 000125 |
| 580 | 77 | 000171 | 610 | 84 | 00017E | 630 | 87 | 0001CD | 635 | 89 | 0001DC |
| 640 | 91 | 0001E0 | 650 | 93 | 0001E4 | 700 | 96 | 0001EA | 750 | 98 | 0001F0 |
| 800 | 100 | 000204 | 810 | 102 | 00021A | 820 | 104 | 000230 | 830 | 106 | 000243 |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*  SOURCE STATEMENTS = 107, PROGRAM SIZE = 38996, SURPROGRAM NAME = MAIN

*STATISTICS*  NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******  276K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

```
 ISN 0002        SUBROUTINE TSRTHS(
           C                                TAGSORT USING HIBBARD'S SHELL SORT
                 O INTAG,
           C                                INTGER ARRAY OF TAGS (POINTERS) SORTED BY
           C                                INTEGER KEY
                 *
                 I KRECRD,
           C                                ARRAY OF INTEGER RECORDS TO SORT
                 ( NWIREC,
           C                                NUMBER OF WORDS IN ONE RECORD
                 ( NRECS,
           C                                NUMBER OF RECORDS (ALSO NUMBER OF TAGS)
                 I KEYLOC,
           C                                LOCATION WITHIN RECORD OF SORT KEY
                 I KSORDR)
           C                                KEY SORT ORDER 'A' FOR ACENDING, 'D' FOR
           C                                DECENDING
           C     -----------------------------------------------------------------
           C
           C HISTORY
           C -------
           C
           C     M A TOMPKINS      LEMSCO  01/27/81         ORIGINAL CODE
           C
           C
           C METHOD
           C ------
           C
           C     CHECK FOR VALID ARGUMENTS.
           C     SORT TAGS IN SPECIFIED KEY SEQUENCE USING HIBBARD'S IMPROVED
           C     SHELLSORT (CACM ALGORITHM # 201)
           C
           C MACHINE DEPENDENT CODE
           C ----------------------
           C
           C     NONE
           C
           C EXTERNAL REFERENCES
           C -------------------
           C
           C     INTEGER IVALUE         ALLOWS THE USE OF LITERALS
           C
           C EXCEPTIONS
           C ----------
           C
           C     THE FOLLOWING INVALID ARGUMENT VALUES PREVENT SORTING AND CAUSE
           C     INTAG(I) TO BE SET TO 0:
           C        NWIREC < 1
           C        NRECS < 1
           C        SORT KEY WHOLLY OR PARTIALLY OUTSIDE RECORD
           C        KSORDR NOT 'A' OR 'D'
           C
           C LOCAL DECLARATIONS
           C ------------------
           C
 ISN 0003        INTEGER INTAG(1)
           C                                INTEGER ARGUMENT
 ISN 0004        INTEGER KRECRD(NWIREC,NRECS)
           C                                ARRAY ARGUMENT
 ISN 0005        INTEGER KDIST
           C                                COMPARISON DISTANCE
 ISN 0006        INTEGER KRN
           C                                PRIMARY COMPARISON RECORD NUMBER
 ISN 0007        INTEGER KRNDIS
           C                                DISTANT COMPARISON RECORD NUMBER
 ISN 0008        INTEGER KTN
           C                                TAG NUMBER OF PRIMARY COMPARISON RECORD
 ISN 0009        INTEGER KTMAX
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)       TSRTHS            OS/360  FORTRAN H EXTENDED            DATE 81.113/16.58.46       PAGE   2
                  C                                 MAXIMUN TAG NUMBER OF PCP             TSP00740
      ISN 0010          INTEGER KTNHI                                                     TSP00750
                  C                                 CURRENT HIGHEST TAG NUMBER OF PCR     TSP00760
                  C                                                                       TSP00770
                  C                                                                       TSP00780
                  C PROCEDURE                                                             TSP00790
                  C ---------                                                             TSP00800
                  C                                                                       TSP00810
                  C                                                                       TSP00820
                  C CHECK ARGUMENTS. SET FLAG FOR BAD ARGUMENT(S).                        TSP00830
                  C                                                                       TSP00840
      ISN 0011          INTAG(1) = 0                                                      TSP00850
                  C                                                                       TSP00860
      ISN 0012          IF(KSORDR.NE.IVALUE('A').AND.KSORDR.NE.IVALUE('D')) GO TO 900     TSP00870
                  C                                                                       TSP00880
      ISN 0014          IF(NWIREC.LT.1) GO TO 900                                         TSP00890
      ISN 0016          IF(NRECS.LT.1) GO TO 900                                          TSP00900
                  C                                                                       TSP00910
      ISN 0018          IF(KEYLOC.LT.1) GO TO 900                                         TSP00920
      ISN 0020          IF(KEYLOC.GT.NWIREC) GO TO 900                                    TSP00930
                  C                                                                       TSP00940
                  C                                                                       TSP00950
                  C INITIALIZE TAGS                                                       TSP00960
                  C                                                                       TSP00970
      ISN 0022          DO 160 KRN = 1,NRECS                                              TSP00980
      ISN 0023             INTAG(KRN) = KRN                                               TSP00990
      ISN 0024      160 CONTINUE                                                          TSP01000
                  C                                                                       TSP01010
                  C                                                                       TSP01020
                  C PERFORM SORT                                                          TSP01030
                  C                                                                       TSP01040
      ISN 0025          KDIST = NRECS                                                     TSP01050
                  C                                                                       TSP01060
                  C WHILE KDIST <> 1 DO                                                   TSP01070
                  C                                                                       TSP01080
      ISN 0026      200 IF(KDIST.EQ.1) GO TO 900                                          TSP01090
      ISN 0028          KDIST = 2*((KDIST + 2)/4) - 1                                     TSP01100
      ISN 0029          KTNMAX = NRECS - KDIST                                            TSP01110
      ISN 0030          DO 400 KTNHI = 1,KTNMAX                                           TSP01120
      ISN 0031             KTN = KTNHI                                                    TSP01130
      ISN 0032      250    IF(KTN.LT.1) GO TO 300                                         TSP01140
      ISN 0034                KRN = INTAG(KTN)                                            TSP01150
      ISN 0035                KRNDIS = INTAG(KTN + KDIST)                                 TSP01160
      ISN 0036                IF((KSORDR.EQ.IVALUE('A')).AND.                             TSP01170
                    &            (KRECRD(KEYLOC,KRN).LE.KRECRD(KEYLOC,KRNDIS))) GO TO 400 TSP01180
      ISN 0038                IF((KSORDR.EQ.IVALUE('D')).AND.                             TSP01190
                    &            (KRECRD(KEYLOC,KRN).GE.KRECRD(KEYLOC,KRNDIS))) GO TO 400 TSP01200
      ISN 0040                INTAG(KTN) = KRNDIS                                         TSP01210
      ISN 0041                INTAG(KTN + KDIST) = KRN                                    TSP01220
      ISN 0042                KTN = KTN - KDIST                                           TSP01230
      ISN 0043             GO TO 250                                                      TSP01240
      ISN 0044      300    CONTINUE                                                       TSP01250
      ISN 0045      400 CONTINUE                                                          TSP01260
      ISN 0046          GO TO 200                                                         TSP01270
                  C                                                                       TSP01280
                  C                                                                       TSP01290
                  C                                                                       TSP01300
      ISN 0047      900 RETURN                                                            TSP01310
      ISN 0048          END                                                               TSP01320

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
KRN      0006  0022  0023  0023  0034  0036  0038  0041
KTN      0008  0031  0032  0034  0035  0040  0041  0042  0042
INTAG    0002  0003  0011  0023  0034  0035  0040  0041
KDIST    0005  0025  0026  0028  0028  0029  0035  0041  0042
KTMAX    0009
KTNHI    0010  0030  0031
NRECS    0002  0004  0016  0022  0025  0029
IVALUE   0012  0012  0036  0038
KEYLOC   0002  0018  0020  0036  0036  0038  0038
KRECRD   0002  0004  0036  0036  0038  0038
KRNDIS   0007  0035  0036  0038  0040
KSORDR   0002  0012  0012  0036  0038
KTNMAX   0029  0030
NWIREC   0002  0004  0014  0020
TSRTHS   0002
```

ORIGINAL PAGE IS
OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES | | | | | |
|---|---|---|---|---|---|---|---|
| 160 | 0024 | 0022 | | | | | |
| 200 | 0026 | 0046 | | | | | |
| 250 | 0032 | 0043 | | | | | |
| 300 | 0044 | 0032 | | | | | |
| 400 | 0045 | 0030 | 0036 | 0038 | | | |
| 900 | 0047 | 0012 | 0014 | 0016 | 0018 | 0020 | 0026 |

/ TSRTHS /  SIZE OF PROGRAM 00035C HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| KRN | SF | I*4 | 0000C0 | KTN | SF | I*4 | 0000C4 | INTAG | SF XR | I*4 | 000000 | KDIST | SF | I*4 | 0000C8 |
| KTMAX | | I*4 | NR | KTNMI | SF | I*4 | 0000CC | NRECS | F | I*4 | 0000D0 | IVALUE | XF | I*4 | 000000 |
| KEYLOC | | I*4 | 0000D4 | KRECRD | XP | I*4 | 000000 | KWNDIS | SF | I*4 | 0000D8 | KSORDR | | I*4 | 00000C |
| KTNMAX | SF | I*4 | 0000E0 | NWIREC | | I*4 | 0000E4 | TSRTHS | | P*4 | 0000E9 | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 160 | 24 | 000174 | 200 | 26 | 000182 | 250 | 32 | 0001H8 | 300 | 44 | 0002CA |
| 400 | 45 | 0002CA | 900 | 47 | 0002E0 | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100001 | 2 | 000104 | 200001 | 13 | 00012E | 100002 | 14 | 000140 | 100003 | 16 | 000148 |
| 100004 | 18 | 000150 | 100005 | 20 | 000158 | 100006 | 22 | 000164 | 100007 | 23 | 000166 |
| 100008 | 25 | 00017E | 100009 | 28 | 000188 | 100010 | 31 | 0001R0 | 100011 | 34 | 0001C0 |
| 100012 | 38 | 000240 | 100013 | 40 | 00029E | 100014 | 46 | 0002DC | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*  SOURCE STATEMENTS = 47, PROGRAM SIZE = 860, SUBPROGRAM NAME =TSRTHS

*STATISTICS*  NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******  296K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

# APPENDIX D

```
REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

                 C
                 C     PROGRAMMER---------------------HADULAR 8/5/80
                 C     CONVERTED TO IBM 3031---------C. HORTON, LOCKHEED, 2/20/81
                 C
                 C     THIS PROGRAM USING THE CROP DEVELEPMENT PROFILE IN
                 C     GREENESS AND LOOKS AT THE COEFFICENTS ALPHA, BETA & T0.
                 C
                 C     DOCUMENTATION CONVENTIONS:
                 C        (VARIABLE)  VARIABLE NAME
                 C        -ROUTINE-   SUBPROGRAM NAME
                 C        /BLOCK/     COMMON BLOCK NAME
                 C
                 C     PROGRAM CONTAINS 4 SECTIONS
                 C       SECTION 1--READS DATE, USER DEFINED DATA, CALCULATES DATES
                 C         AS DAYS OF YEAR, CALCULATES SINES OF SUN ANGLES,
                 C         FINDS POSSIBLE DUPLICATED ACQUISITION DATE (KREP)
                 C         AND PRINTS REPORT HEADER
                 C
                 C       SECTION 2--READS FILE HEADER & TRAINING FLD COORDINATES -RDFLES-,
                 C         READS TRAINING FLD, SELECTS PIXELS FROM TRAINING FLD,
                 C         PRINTS TRAINING FIELD COORDINATES ON REPORT,
                 C         CALLS -SDEV- TO CALCULATE MEAN (XMNTRN) AND ST. DEV. (SGMATR)
                 C         FOR PIXELS IN TRAINING FLD AND CALLS -TRJXM- TO CALCULATE
                 C         A (TRAINA), ALPHA (TRALPH), BETA (TRBETA) AND T0 (TRT0)
                 C         FOR PIXELS IN THE TRAINING FIELD
                 C
                 C       SECTION 3--CREATES AND WRITES HEADER
                 C
                 C       SECTION 4--REREADS FILE HEADER AND READS TEST FLD COORDINATES,
                 C         READS AND PROCESSES TEST FLD USING CONSTANTS CALCULATED
                 C         IN SECTION 2 (XMNTRN, SGMATR, TRAINA, TRALPH, TRBETA, TRT0) AND
                 C         CALLS -TRJXM- TO CALCULATE ALPHA (IALPHF), BETA (IBETBF),
                 C         T0 (IT0BUF) AND SIGMA (ICHIBF) FOR EACH PIXEL IN THE TEST FL
                 C         WRITES COEFFICIENTS CALCULATED TO FILE 8
                 C
                 C     ATTACHMENT A:  FILE CLASFYG CONSTANT--FILE OF VARIABLES & DECRIPTIONS
                 C
                 C
                 C     ****************************************************************************
                 C     ****************************************************************************
                 C
ISN 0002               IMPLICIT INTEGER*4 (C-H)
                 C
                 C     IDATA HAS (72 + 4*196) CHARACTERS FOR A MAX OF 8 ACQUISITIONS
ISN 0003               COMMON /LINES/ NXTLNE, LINENO, IDATA(856,8), NOACQS
ISN 0004               LOGICAL*1 IDATA,IHOLD1(4)
ISN 0005               EQUIVALENCE (IHOLD1(1),IHOLD4)
                 C
ISN 0006               COMMON /MISC/NOACQ, ACQDT(8), KSEGM,DATE(4),LANDST(8)
ISN 0007               REAL*8 ACQDT
ISN 0008               LOGICAL*1 ACQDTH(8,8)
ISN 0009               EQUIVALENCE (ACQDT(1),ACQDTH(1,1))
                 C
ISN 0010               COMMON /USED/ ICROP(4), ZX(4), SANGGG(8)
ISN 0011               INTEGER*4 SANGGG
ISN 0012               DIMENSION SANG(4)
                 C
ISN 0013               REAL*8 WEIGHT(8), TODAY
ISN 0014               INTEGER*2 STIME(2), NTIME(2)
ISN 0015               DIMENSION ICHBUF(196,4,8)
ISN 0016               REAL*4 CLNDST(2,4)
ISN 0017               DIMENSION XDAY(8)
ISN 0018               DIMENSION DELTX(8)
ISN 0019               DIMENSION VERTPX(4),VERTLN(4),XGREEN(8,10)
ISN 0020               DIMENSION YGREEN(10),SGMATR(8),IDATE(8)
ISN 0021               DIMENSION SIGMA(4), DELTA(4), XMNTRN(8), APIXGR(10)
ISN 0022               INTEGER*4 IALPHF(196), IBETBF(196), IT0BUF(196), ICHIBF(196)
                 C
ISN 0023               LOGICAL*1 IHOBF(3060), BSEG(4), IDAY, IMON, IYEAR
ISN 0024               EQUIVALENCE (BSEG(1),JSEGM)
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0025            LOG(CAL*) KA,KB,KC,KD,KE,KF,KG,KH,KI,KJ,KK,KL,KM,KN,KO,KP,KR,KS                CLA00740
ISN 0026            LOG(CAL*) KT,KU,KV                                                             CLA00750
ISN 0027            LOG(CAL*) KFORMAL, KBLANK                                                      CLA00760
                    C                                                                              CLA00770
ISN 0028            DATA KA/'A'/,KB/'B'/,KC/'C'/,KD/'D'/,KE/'E'/,KF/'F'/,KG/'G'/                   CLA00780
ISN 0029            DATA KH/'H'/,KI/'I'/,KJ/'J'/,KK/'K'/,KL/'L'/,KM/'M'/,KN/'N'/                   CLA00790
ISN 0030            DATA KO/'O'/,KP/'P'/,KQ/'Q'/,KS/'S'/,KT/'T'/,KU/'U'/                           CLA00800
ISN 0031            DATA  THOLD4/8/, KBLANK/' '/                                                   CLA00810
                    C                                                                              CLA00820
ISN 0032            DATA CLNDST(1,1)/1./,CLNDST(1,2)/1./,CLNDST(1,3)/1./                           CLA00830
ISN 0033            DATA CLNDST(1,4)/1./                                                           CLA00840
ISN 0034            DATA CLNDST(2,1)/1.161/, CLNDST(2,2)/1.230/, CLNDST(2,3)/1.246/                CLA00850
ISN 0035            DATA CLNDST(2,4)/1.062/                                                        CLA00860
                    C                                                                              CLA00870
ISN 0036            DATA XTOL/3.E-05/, MODE/1/                                                     CLA00880
ISN 0037            DATA LIMIT/8/                                                                  CLA00890
ISN 0038            DATA K1/8/,K2/16/,K106/78/,K150/104/,K200/128/,K304/196/                        CLA00900
                    C                                                                              CLA00910
                    C ***************************************************************************CLA00920
                    C ***************************************************************************CLA00930
                    C                                                                              CLA00940
                    C  SECTION 1--READS DATE, USER DEFINED DATA, CALCULATES DATES AS DAYS          CLA00950
                    C     OF YEAR, CALCULATES SINES OF SUN ANGLES,                                 CLA00960
                    C     FINDS POSSIBLE DUPLICATED ACQUISITION DATES (KREP) AND                   CLA00970
                    C     PRINTS HEADING AT TOP OF REPORT                                          CLA00980
                    C                                                                              CLA00990
                    C  READ DATE AND TIME                                                          CLA01000
ISN 0039            CALL CPTIME (TODAY, STIME, ICPTIM, IVCPTM)                                     CLA01010
                    C                                                                              CLA01020
                    C  CHANGE SYSTEM ERROR ROUTINE TO NO MESSAGE, NO ABORT, ONLY COUNT AT          CLA01030
                    C  END OF PROGRAM  FOR OVERFLOW, UNDERFLOW AND EXPONENT TOO LARGE              CLA01040
ISN 0040            CALL ERRSET (207, 300, -1, 1, 0, -1)                                           CLA01050
ISN 0041            CALL ERRSET (209, 300, -1, 1, 0, -1)                                           CLA01060
ISN 0042            CALL ERRSET (252, 300, -1, 1, 0, -1)                                           CLA01070
                    C                                                                              CLA01080
                    C  READ USER DEFINITION CARDS                                                  CLA01090
ISN 0043            CALL RDEFC                                                                     CLA01100
                    CCCC  WRITE (3,9999) NOACQ, KSEGM,(ACQDT(I),I=1,8),(SANGGG(I),I=1,8),          CLA01110
                    CCCC 1   (LANDST(I),I = 1,8)                                                   CLA01120
ISN 0044            9999  FORMAT (' NOACQ,KSEGM,ACQDT,SANG,LANDST=',/,I4,1X,A4,                   CLA01130
                          1  1X,4A7,/,1X,8I3,1X,8I2)                                              CLA01140
                    C                                                                              CLA01150
                    C  CALL CROPG SUBROUTINE                                                       CLA01160
ISN 0045            CALL CROPG(IERROR)                                                             CLA01170
                    C                                                                              CLA01180
                    C  IF ALL SUNANGLES ARE THE SAME, STOP PROGRAM                                 CLA01190
ISN 0046            DO 6 I = 2,NOACQ                                                               CLA01200
ISN 0047            IF (SANGGG(I) .NE. SANGGG(1) ) GO TO 8                                         CLA01210
ISN 0049          6 CONTINUE                                                                       CLA01220
                    C                                                                              CLA01230
                    C  ALL SUNANGLES ARE THE SAME, HALT EXECUTION                                  CLA01240
ISN 0050            WRITE (3,7)   (SANGGG(I),I=1,NOACQ)                                            CLA01250
ISN 0051            WRITE (6,7)   (SANGGG(I),I=1,NOACQ)                                            CLA01260
ISN 0052            WRITE (3,9)                                                                    CLA01270
ISN 0053            WRITE (6,9)                                                                    CLA01280
ISN 0054          7   FORMAT (' SUNANGLES FROM DATA BASE =',8I4)                                   CLA01290
ISN 0055          9   FORMAT (' USER SHOULD INPUT NEW SUNANGLES IN USERG FILE.')                   CLA01300
ISN 0056            STOP                                                                           CLA01310
                    C                                                                              CLA01320
ISN 0057          8   IF (IERROR .NE. 0) STOP                                                      CLA01330
                    CCCC  WRITE (3,9998) ICROP, IX                                                 CLA01340
ISN 0059            9998  FORMAT (' ICROP, IX=',2X,4A1, 4F8.2)                                    CLA01350
ISN 0060            NOACQS = NOACQ                                                                 CLA01360
                    C                                                                              CLA01370
                    C  CALCULATE SUN ANGLE VALUE FROM SUN ANGLE GIVEN                              CLA01380
ISN 0061            DO 3  I = 1,8                                                                  CLA01390
ISN 0062            SAG(I) = 1.0                                                                   CLA01400
ISN 0063            IF (SANGGG(I) .EQ. 0) GO TO 3                                                  CLA01410
ISN 0065            ANG = SANGGG(I) * 0.01745                                                      CLA01420
ISN 0066            IF (ANG .LE. 0.02) ANG = 20.0 * 0.01745                                        CLA01430
ISN 0068            SAG(I) = 0.76604 / SIN(ANG)                                                    CLA01440
ISN 0069          3   CONTINUE                                                                     CLA01450
                    CCCC  WRITE (3,9997) (SAG(I),I = 1,8)                                          CLA01460
ISN 0070            9997  FORMAT (' SUN ANGLE CORRECTION FACTOR = ',/,4F9.5)                       CLA01470
                    C                                                                              CLA01480
                    C  CONVERT ACQUISITION DATES FROM ALPHA CODE TO COMPUTATIONAL NUMBER            CLA01490
ISN 0071            READ(5) 10                                                                     CLA01500
ISN 0072            WRITE (30,17) (ACQDT(III),III = 1,NOACQ)                                       CLA01510
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0073          REWIND 30
ISN 0074          READ (30,18) ((DATE(I,II),II = 1,NOACQ))
ISN 0075       17   FORMAT (8A4)
ISN 0076       18   FORMAT (5(2X,I3,3X))
ISN 0077          DO 5 I = 1,4
ISN 0078          XD(Y(I) = IDATE(I)/100.
ISN 0079        5 CONTINUE
                C
                C   CHECK TO SEE IF THE DAYS ARE REPEATED
ISN 0080          KREP = 0
ISN 0081          DO 1 JM = 2,NOACQS
ISN 0082          IF(IDATE(JM).EQ.IDATE(JM-1))  KREP=JM
ISN 0084        1 CONTINUE
                C
                C
                C  PRINT HEADING ON PRINTER REPORT
ISN 0085          WRITE(6,6000) IODAY,STIME(1),STIME(2),KSEGM,ICROP
ISN 0086       6000 FORMAT(1H1,10X,'SUPPORTING RESEARCH MULTITEMPORAL '
                  *'CLASSIFICATION'//11X,'PROCESSING DATE - ',A8,' AT ',I2,':',I2//
                  *11X,'SEGMENT NUMBER - ',A4,5X,'CROP OF INTEREST - ',4A1)
                C
                C
                C  ************************************
                C  SECTION 2--READS FILE HEADER & TRAINING FLD COORDINATES.
                C     PRINTS TRAINING FLD COORDINATES.
                C     READS THE TRAINING FLD AND CALCULATES THE FOLLOWING VALUES
                C      XMNTPH--MEAN FROM SDEV OF PIXELS IN TRAINING FIELD
                C      SGMATR--ST. DEV. FROM SDEV OF PIXELS IN TRAINING FIELD
                C      A--CALCULATED A VALUE FROM -TRJXM- FOR TRAINING FIELD
                C      TRALPH--CALCULATED ALPHA VALUE FROM TRJXM FOR TRAINING FL
                C
ISN 0087       100  ITYPE = 1
ISN 0088          CALL RDFLES(ITYPE, IHEDNE, VERTPX, VERTLN, IERROR)
                CCCC  IF (IERROR .NE. 0) STOP
                C
ISN 0089          WRITE(6,6002) ((VERTLN(I),VERTPX(I)),I = 1,4)
ISN 0090       6002 FORMAT(1H0,10X,'TRAINING FIELD -  LINE NO.     SAMP. NO.'/
                  *4(30X,F5.1,7X,F5.1/))
                C
                C   DETERMINE THE MAX AND MIN LINE NO. AND PIXEL NO. OF TEST FLD
                C
ISN 0091          LMAX = 1
ISN 0092          LMIN = 196
ISN 0093          MINPXL = 196
                C  CHECK CORNERS OF USER SUPPLIED TRAINING FLD
ISN 0094          MAXPXL = 1
ISN 0095          DO 24 JM = 1,4
ISN 0096          LINENO = VERTLN(JM)
ISN 0097          IPIXEL = VERTPX(JM)
ISN 0098          IF(LINENO.GE.LMAX) LMAX = LINENO
ISN 0100          IF(LINENO.LE.LMIN) LMIN = LINENO
ISN 0102          IF(IPIXEL.GE.MAXPXL) MAXPXL = IPIXEL
ISN 0104          IF(IPIXEL.LT.MINPXL) MINPXL = IPIXEL
                C
ISN 0106       24 CONTINUE
                C
                C
                C   **********  NORMAL PROCESSING OF TEST FIELD  **********
                C
                C  SET NUMBER OF PIXELS SELECTED TO 0
ISN 0107          NOTYPX = 0
                C
                C  ****************  PROCESS INDIVIDUAL LINES  ****************
                C
ISN 0108          DO 78 KOPLNE = LMIN, LMAX
ISN 0109          LINENO = KOPLNE
                C
                C   READ DATA FOR ONE LINE OF PIXELS
ISN 0110          ITYPE = 2
ISN 0111          CALL RDFLES (ITYPE, IDATA, VERTPX, VERTLN, IERROR)
                CCCC  IF (IERROR .NE. 0) STOP
                C
                C  MOVE PIXEL RADIANCE VALUES FROM THE 5 TO 4 RADIANCE FILES
                C    NUMBER OF FILES DEPENDS ON NUMBER OF ACQUISITIONS
ISN 0112          DO 50 KORACQ = 1,NOACQ
ISN 0113          DO 49 KORCHN = 1,4
ISN 0114          INCRF = (KORCHN - 1) * 196
                C
```

CLA01520 CLA01530 CLA01540 CLA01550 CLA01560 CLA01570 CLA01580 CLA01590 CLA01600 CLA01610 CLA01620 CLA01630 CLA01640 CLA01650 CLA01660 CLA01670 CLA01680 CLA01690 CLA01700 CLA01710 CLA01720 CLA01730 CLA01740 CLA01750 CLA01760 CLA01770 CLA01780 CLA01790 CLA01800 CLA01810 CLA01820 CLA01830 CLA01840 CLA01850 CLA01860 CLA01870 CLA01880 CLA01890 CLA01900 CLA01910 CLA01920 CLA01930 CLA01940 CLA01950 CLA01960 CLA01970 CLA01980 CLA01990 CLA02000 CLA02010 CLA02020 CLA02030 CLA02040 CLA02050 CLA02060 CLA02070 CLA02080 CLA02090 CLA02100 CLA02110 CLA02120 CLA02130 CLA02140 CLA02150 CLA02160 CLA02170 CLA02180 CLA02190 CLA02200 CLA02210 CLA02220 CLA02230 CLA02240 CLA02250 CLA02260 CLA02270 CLA02280 CLA02290

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0115          DO 49 KURPXL = MINPXL, MAXPXL                                  CLA02300
                C SKIP OVER FIRST 72 CHARACTERS OF LEAD DATA ON TAPE RECORD      CLA02310
ISN 0116          LOC = INCRE + KURPXL + 72                                      CLA02320
ISN 0117          IHOLD(4) = IDATA(LOC+KURACQ)                                   CLA02330
ISN 0118          ICHNPX(KURPXL,KURCHN,KURACQ) = IHOLD4                          CLA02340
ISN 0119     49   CONTINUE                                                       CLA02350
ISN 0120     50   CONTINUE                                                       CLA02360
                C                                                               CLA02370
                C                                                               CLA02380
                C ******  PROCESS INDIVIDUAL PIXELS IN THE CURRENT LINE  ******  CLA02390
ISN 0121          DO 77 KURPXL = MINPXL, MAXPXL                                  CLA02400
ISN 0122          IPIXEL = KURPXL                                                CLA02410
                C                                                               CLA02420
                C  CHECK IF LINE NO. AND PIXEL NO. LIE WITHIN THE TRAINING FIELD CLA02430
ISN 0123          XLINE = LINENO                                                 CLA02440
ISN 0124          XPIXEL = IPIXEL                                                CLA02450
ISN 0125          CALL RECT (XLINE,XPIXEL,VERTLN,VERTPX,ICOR)                    CLA02460
ISN 0126          IF(ICOR.EQ.1) GO TO 77                                         CLA02470
                C                                                               CLA02480
                C PIXEL IN TRAINING FIELD                                       CLA02490
                C                                                               CLA02500
                C SKIP OVER SCREENED PIXELS                                     CLA02510
ISN 0128          DO 24 J = 1,NOACQS                                             CLA02520
ISN 0129          IF (ICHNPX(IPIXEL,4,J) .GT. 127) GO TO 77                      CLA02530
ISN 0131     24   CONTINUE                                                       CLA02540
                C                                                               CLA02550
                C INCRE COUNT OF SELECTED PIXELS (IN USER TRAINING FLD, NOT SCREENED) CLA02560
ISN 0132          NOTRPX=NOTRPX+1                                                CLA02570
                C                                                               CLA02580
                CCCC  WRITE (6,9996) LINENO, IPIXEL                             CLA02590
ISN 0133        9996  FORMAT (' LINENO=',I4,'  PIXEL=',I4)                      CLA02600
                C   COMPUTE THE GREENESS VALUES AND STORE IN XGREEN FOR TRAINING FLD PIXCLA02610
                C                                                               CLA02620
ISN 0134          DO 27 J = 1,NOACQS                                             CLA02630
ISN 0135          INDEX = LANDST(J) - 1                                          CLA02640
ISN 0136          XGREEN(J,NOTRPX) = (-.29 * ICHNPX(IPIXEL,1,J) * CLNDST(INDEX,1) - CLA02650
                 1              .562 * ICHNPX(IPIXEL,2,J) * CLNDST(INDEX,2) +  CLA02660
                 2              .6 * ICHNPX(IPIXEL,3,J) * CLNDST(INDEX,3) +    CLA02670
                 3  .2455*2.0*ICHNPX(IPIXEL,4,J) * CLNDST(INDEX,4) ) * SANG(J) + 10.CLA02680
                CCCC  WRITE (6,9995) J, (ICHNPX(IPIXEL,I,J),I = 1,4),XGREEN(J,NOTRPX) CLA02690
ISN 0137        9995  FORMAT (' ACQ',I1,' VALUES=',4I6,' GREEN=',F12.6)         CLA02700
ISN 0138     27   CONTINUE                                                       CLA02710
                C                                                               CLA02720
                C END OF TRAINING PIXEL COLLECTION WHEN 10 PIXELS COLLECTED     CLA02730
ISN 0139          IF (NOTRPX .EQ. 10) GO TO 28                                   CLA02740
                C                                                               CLA02750
                C END OF PIXEL DO LOOP                                          CLA02760
ISN 0141     77   CONTINUE                                                       CLA02770
                C                                                               CLA02780
                C                                                               CLA02790
                C END OF LINE DO LOOP                                           CLA02800
ISN 0142     78   CONTINUE                                                       CLA02810
                C                                                               CLA02820
                C                                                               CLA02830
                C                                                               CLA02840
                C GO TO PROCESS TEST FLD IF AT LEAST 2 ACCEPTABLE PIXEL IN TRAINING FLDCLA02850
ISN 0143     28   IF(NOTRPX.GT.1) GO TO 48                                       CLA02860
                C                                                               CLA02870
                C  STOP PROCESSING IF LESS THAN 2 ACCEPTABLE PIXELS IN TRAINING FLD CLA02880
ISN 0145          WRITE(6,950) NOTRPX                                            CLA02890
ISN 0146     950    FORMAT(1H0,'PROGRAM TERMINATED BECAUSE THERE WERE LESS'/     CLA02900
                *' THAN TWO PIXELS IN THE TEST FIELD.  NOTRPX = ',I1//)         CLA02910
ISN 0147          WRITE (6,9001)                                                 CLA02920
ISN 0148    9001  FORMAT (' NOT ENOUGH PIXELS IN TEXT FIELD')                    CLA02930
ISN 0149          STOP                                                           CLA02940
                C                                                               CLA02950
                C                                                               CLA02960
                C                                                               CLA02970
                C *****  ACCEPTABLE SET OF TRAINING PIXELS HAS BEEN COLLECTED  ***** CLA02980
                C                                                               CLA02990
                C COMPUTE MEAN AND STANDARD DEVIATION                           CLA03000
                C                                                               CLA03010
ISN 0150     48   DO 29 K = 1,NOACQS                                             CLA03020
ISN 0151          DO 31 I = 1,NOTRPX                                             CLA03030
ISN 0152          YGREEN(I) = XGREEN(K,I)                                        CLA03040
ISN 0153     31   CONTINUE                                                       CLA03050
                C                                                               CLA03060
ISN 0154          CALL SDEV (YGREEN, NOTRPX, XMEAN, SIGMA)                       CLA03070
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0155            XMNTRN(M) = XMEAN                                          CLA03040
ISN 0156            SGMATR(M) = SIGMAG                                         CLA03090
ISN 0157            IF(M.EQ.XMEP) SGMATR(M) = 20.*SGMATR(M)                    CLA03100
ISN 0159         24 CONTINUE                                                   CLA03110
                 C                                                             CLA03120
                 C  PRINT MEAN AND STD. DEV. FOR TRAINING FIELD.               CLA03130
                 C                                                             CLA03140
ISN 0160            WRITE (6,6003) NOTRPX, (ACQDT(I),I = 1,NOACQ)              CLA03150
ISN 0161         6003 FORMAT(1H ,10X,'MEANS AND STD. DEV. FOR TRAINING FIELD',  CLA03160
                    *' BASED ON ',I2,' PIXELS -'//15X,'GREENESS',16X,           CLA03170
                    *'----------ACQUISITION DATES----------'/15X,'NUMBER',17X,  CLA03180
                    *  8(A5,3X))                                               CLA03190
                 C                                                             CLA03200
ISN 0162            I = 1                                                      CLA03210
ISN 0163            WRITE(6,6004) I,XMNTRN,SGMATR                              CLA03220
ISN 0164         6004 FORMAT(1H0,18X,I1,6X,'MEAN',5X,8F8.2/26X,'STD. DEV.',8F8.2) CLA03230
ISN 0165            WRITE(6,6005)                                              CLA03240
ISN 0166         6005 FORMAT(1H0,10X,'CONSTANTS FOR MODEL -'//16X,'CHANNEL'/    CLA03250
                    *16X,'NUMBER',16X,'A',12X,'ALPHA',11X,'BETA',11X,'TO',12X,'CHISQ') CLA03260
                 C                                                             CLA03270
ISN 0167            WRITE(6,6006) I,ZX                                         CLA03280
ISN 0168         6006 FORMAT(1H0,18X,I1,6X,'INITIAL',1X,4(F8.2,7X))            CLA03290
                 C                                                             CLA03300
                 C  COMPUTE GREENESS FIT FOR TRAINING FIELD                    CLA03310
ISN 0169            XLAMDA = 1.E-03                                            CLA03320
ISN 0170            XCHIQ = 200.                                               CLA03330
ISN 0171            NMODE = 0                                                  CLA03340
ISN 0172            NTERM=4                                                    CLA03350
                 C                                                             CLA03360
                 C  CALCULATE WEIGHT FACTOR                                    CLA03370
ISN 0173            DO 2999 I = 1,NOACQS                                       CLA03380
ISN 0174         2999 WEIGHT(I) = 1. / (SGMATR(I) * SGMATR(I) )                CLA03390
CCCC                WRITE (3,9994) WEIGHT, XDAY, SANGGG,SANG                   CLA03400
ISN 0175         9994 FORMAT (' WEIGHT=',8F7.2,/,'  DOY=',8F6.1,/,' SANGGG=',8I5,/, CLA03410
                    1  '  SANG=',8F7.2)                                       CLA03420
                 C                                                             CLA03430
                 C  FIT MODEL TO THE SELECTED TRAINING FLD PIXELS.  ITERATE UNTIL AN CLA03440
                 C  ACCEPTABLE FIT OBTAINED OR A MAXIMUM OF 24 ITERATIONS      CLA03450
ISN 0176            DO 3001 IA = 1,24                                          CLA03460
                 C                                                             CLA03470
                 C  IPRINT USED IN DEBUG TO TRACK VALUE OF A(1). -1=PRINT      CLA03480
ISN 0177            IPRINT = -1                                                CLA03490
                 C                                                             CLA03500
ISN 0178            CALL TRJXM(XDAY,XMNTRN,SGMATR,NOACQS,NTERM,MODE,ZX,PELTA,SIGMA, CLA03510
                    1 XLAMDA,OFIX,XISQR,NMODE, WEIGHT, IPRINT)                 CLA03520
ISN 0179            XDIF = XCHIN-XISQR                                         CLA03530
ISN 0180            XTH = XISQR*XTOL                                           CLA03540
ISN 0181            IF(ABS(XDIF).LE.XTH) GO TO 3002                            CLA03550
ISN 0183            XCHIN = XISQR                                              CLA03560
ISN 0184         3001 CONTINUE                                                 CLA03570
                 C                                                             CLA03580
ISN 0185         3002   CONTINUE                                               CLA03590
                 C                                                             CLA03600
                 C  PRINT MODEL PARAMETERS, TOLERANCES AND CHI SQ.             CLA03610
ISN 0186            WRITE(6,6007) (ZX(K),SIGMA(K),K = 1,NTERM),XISQR           CLA03620
ISN 0187         6007 FORMAT(1H ,25X,'FINAL  ',1X,4(F8.2,'+-',F5.2),1X,F8.2)   CLA03630
                 C                                                             CLA03640
                 C  SAVE MODEL PARAMETERS TO USE FOR TEST FIELD                CLA03650
ISN 0188            TRANA = ZX(1)                                              CLA03660
ISN 0189            TRALPH = ZX(2)                                             CLA03670
ISN 0190            TRBETA = ZX(3)                                             CLA03680
ISN 0191            TRTO = ZX(4)                                               CLA03690
                 C                                                             CLA03700
                 C  ************************************************************CLA03710
                 C  ************************************************************CLA03720
                 C  SECTION 3--CREATE AND WRITE HEADER RECORD                   CLA03730
                 C                                                             CLA03740
                 C                                                             CLA03750
                 C     CLEAR THE HEADER RECORD AWAY.                           CLA03760
ISN 0192            DO 104 I = 1,3040                                          CLA03770
ISN 0193         104 IHEDNE(I) = 0                                             CLA03780
                 C                                                             CLA03790
                 C     LOAD DATA INTO HEADER RECORD.                           CLA03800
                 C                                                             CLA03810
                 C  COMPUTING SYSTEM ID (BYTES 1-12)                           CLA03820
ISN 0194            IHEDNE( 1) = KS                                            CLA03830
ISN 0195            IHEDNE( 2) = KP                                            CLA03840
ISN 0196            IHEDNE( 3) = KBLANK                                        CLA03850
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0197            IHEDHF( 4) = KM                                          CLA03860
ISN 0198            IHEDHF( 5) = KH                                          CLA03870
ISN 0199            IHEDHF( 6) = KL                                          CLA03880
ISN 0200            IHEDHF( 7) = KI                                          CLA03890
ISN 0201            IHEDHF( 8) = KHLANK                                      CLA03900
ISN 0202            IHEDHF( 9) = KT                                          CLA03910
ISN 0203            IHEDHF(10) = KE                                          CLA03920
ISN 0204            IHEDHF(11) = KM                                          CLA03930
ISN 0205            IHEDHF(12) = KP                                          CLA03940
ISN 0206            IHEDHF(13) = KW                                          CLA03950
ISN 0207            IHEDHF(14) = KHLANK                                      CLA03960
ISN 0208            IHEDHF(15) = KC                                          CLA03970
ISN 0209            IHEDHF(16) = KL                                          CLA03980
ISN 0210            IHEDHF(17) = KA                                          CLA03990
ISN 0211            IHEDHF(18) = KF                                          CLA04000
ISN 0212            IHEDHF(19) = KS                                          CLA04010
ISN 0213            IHEDHF(20) = KI                                          CLA04020
ISN 0214            IHEDHF(21) = KF                                          CLA04030
ISN 0215            IHEDHF(22) = KI                                          CLA04040
ISN 0216            IHEDHF(23) = KE                                          CLA04050
ISN 0217            IHEDHF(24) = KW                                          CLA04060
ISN 0218            DO 114 I = 25,52                                         CLA04070
ISN 0219      114   IHFDHF(I) = KHLANK                                       CLA04080
              C  SENSOR ID (1 - CHASSIFICATION FILE)                         CLA04090
ISN 0220            IHEDHF(53) = KI                                          CLA04100
ISN 0221            DO 115 I=54,60                                           CLA04110
ISN 0222      115   IHFDHF(I) = KHLANK                                       CLA04120
              C  CONVERT DATE OF PROCESSING TO LOGICAL*1 VALUES              CLA04130
ISN 0223            REWIND 30                                                CLA04140
ISN 0224            WRITE (30,1114) TODAY                                    CLA04150
ISN 0225      1114  FORMAT (A8)                                              CLA04160
ISN 0226            REWIND 30                                                CLA04170
ISN 0227            READ(30,1115) IMON, IDAY, IYEAR                          CLA04180
ISN 0228      1115  FORMAT (I2,1X,I2,1X,I2)                                  CLA04190
              C DATE OF PROCESSING - DAY                                     CLA04200
ISN 0229            IHEDHF(61) = IDAY                                        CLA04210
              C DATE OF PROCESSING - MONTH                                   CLA04220
ISN 0230            IHEDHF(62) = IMON                                        CLA04230
              C DATE OF PROCESSING - YEAR                                    CLA04240
ISN 0231            IHEDHF(63) = IYEAR                                       CLA04250
              C TAPE SEQUENCE NUMBER                                         CLA04260
ISN 0232            IHEDHF(64) = 1                                           CLA04270
              C  SEGMENT NUMBER                                              CLA04280
ISN 0233            REWIND 30                                                CLA04290
ISN 0234            WRITE (30,1124) KSEGM                                    CLA04300
ISN 0235      1124  FORMAT (A4)                                              CLA04310
ISN 0236            REWIND 30                                                CLA04320
ISN 0237            READ (30,1117) JSEGM                                     CLA04330
ISN 0238      1117  FORMAT (I4)                                              CLA04340
ISN 0239            IHEDHF(65)=RSEG(1)                                       CLA04350
ISN 0240            IHEDHF(66) = RSEG(2)                                     CLA04360
ISN 0241            IHEDHF(67) = RSEG(3)                                     CLA04370
ISN 0242            IHEDHF(68) = RSEG(4)                                     CLA04380
              C CHANNELS ACTIVE (10000000)                                   CLA04390
ISN 0243            IHEDHF(81) = K200                                        CLA04400
              C PROCESSING FLAG (1 - PROCESSED)                              CLA04410
ISN 0244            IHEDHF(89) = 1                                           CLA04420
              C NUMBER OF CHANNELS                                           CLA04430
ISN 0245            IHEDHF(90) = 1                                           CLA04440
              C NUMBER OF BITS PER BYTE                                      CLA04450
ISN 0246            IHEDHF(91) = K10                                         CLA04460
              C BYTE ADD. OF START OF DATA IN SCAN                           CLA04470
ISN 0247            IHEDHF(93) = 1                                           CLA04480
              C NUMBER OF PIXELS PER SCAN.                                   CLA04490
ISN 0248            IHEDHF(97) = K304                                        CLA04500
              C RECORD SIZE IN BYTES (360)                                   CLA04510
ISN 0249            IHEDHF(100) = 1                                          CLA04520
              C  LOW ORDER BYTES OF RECORD SIZE                              CLA04530
ISN 0250            IHEDHF(101) = K104                                       CLA04540
              C NUMBER OF RECORDS TO MAKE DATA SET                           CLA04550
ISN 0251            IHEDHF(104) = 1                                          CLA04560
              C LENGTH OF ANCILLARY BLOCK IN BYTES (70)                      CLA04570
ISN 0252            IHEDHF(106) = K10A                                       CLA04580
              C START PIXEL NUMBER                                           CLA04590
ISN 0253            IHEDHF(109) = 1                                          CLA04600
              C STOP PIXEL NUMBER                                            CLA04610
ISN 0254            IHEDHF(111) = K304                                       CLA04620
              C YOUNG LENGTH OF GENERATING COMPUTER (16)                     CLA04630
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0255            IHEDHF(753) = K20                                          CLA04640
                  C NUMBER OF DATA SETS PER RECORD                             CLA04650
ISN 0256            IHEDHF(1778) = 1                                           CLA04660
                  C NO. OF CHANNELS IN FIRST REC. OF DATA SET                  CLA04670
ISN 0257            IHEDHF(1784) = 1                                           CLA04680
                  C NUMBER OF ELEMENTS PER SCAN                                CLA04690
ISN 0258            IHEDHF(1788) = K304                                        CLA04700
                  C PIXEL SKIP FACTOR (PROCESS ALL PIXELS)                     CLA04710
ISN 0259            IHEDHF(1790) = 1                                           CLA04720
                  C SCAN SKIP FACTOR (PROCESS ALL SCANS)                       CLA04730
ISN 0260            IHEDHF(1792) = 1                                           CLA04740
                  C                                                            CLA04750
                  C     LOAD ACQUISITION DATES INTO HEADER RECORD.             CLA04760
                  C                                                            CLA04770
ISN 0261            DO 116 I = 1,5                                             CLA04780
                  C ACQ. DATE FOR FIRST ACQ.                                   CLA04790
ISN 0262            IHEDHF(2248+I)=ACQDTH(I,1)                                 CLA04800
                  C ACQ. DATE FOR SECOND ACQ.                                  CLA04810
ISN 0263            IHEDHF(2256+I)=ACQDTH(I,2)                                 CLA04820
                  C ACQ. DATE FOR THIRD ACQ.                                   CLA04830
ISN 0264            IHEDHF(2264+I)=ACQDTH(I,3)                                 CLA04840
                  C ACQ. DATE FOR FOURTH ACQ.                                  CLA04850
ISN 0265            IHEDHF(2272+I)=ACQDTH(I,4)                                 CLA04860
                  C ACQ. DATE FOR FIFTH ACQ.                                   CLA04870
ISN 0266            IHEDHF(2280+I)=ACQDTH(I,5)                                 CLA04880
ISN 0267            IF(NOACQS.GE.6) IHEDHF(2288+I)=ACQDTH(I,6)                 CLA04890
ISN 0269            IF(NOACQS.GE.7) IHEDHF(2296+I)=ACQDTH(I,7)                 CLA04900
ISN 0271            IF(NOACQS.EQ.8) IHEDHF(2304+I)=ACQDTH(I,8)                 CLA04910
ISN 0273  116       CONTINUE                                                   CLA04920
                  C                                                            CLA04930
                  C     LOAD DATA FOR PRODUCTION FILM CONVERTER.               CLA04940
                  C                                                            CLA04950
                  C N=1000 SCAN LINES PER FRAME                                CLA04960
ISN 0274            IHEDHF(2759) = 1                                           CLA04970
ISN 0275            DO 117 I = 2760, 2789                                      CLA04980
ISN 0276  117       IHEDHF(I) = KBLANK                                         CLA04990
                  C  JOB ID (BYTES 2760 - 2789)                                CLA05000
ISN 0277            IHEDHF(2760) = KS                                          CLA05010
ISN 0278            IHEDHF(2761) = KP                                          CLA05020
ISN 0279            IHEDHF(2762) = KC                                          CLA05030
ISN 0280            IHEDHF(2764) = KL                                          CLA05040
ISN 0281            IHEDHF(2765) = KA                                          CLA05050
ISN 0282            IHEDHF(2766) = KS                                          CLA05060
ISN 0283            IHEDHF(2767) = KS                                          CLA05070
ISN 0284            IHEDHF(2769) = KF                                          CLA05080
ISN 0285            IHEDHF(2770) = KO                                          CLA05090
ISN 0286            IHEDHF(2771) = KR                                          CLA05100
ISN 0287            DO 118 I = 1,4                                             CLA05110
                  C CROP OF INTEREST FROM INPUT DATA                           CLA05120
ISN 0288  118       IHEDHF(2772+I) = ICROP(I)                                  CLA05130
                  C DATE OF PROCESSING                                         CLA05140
                  C SCAN TYPE (1 - SMOOTHED)                                   CLA05150
ISN 0289            IHEDHF(2795) = 1                                           CLA05160
                  C CAMERA TYPE (1 - 5 INCH)                                   CLA05170
ISN 0290            IHEDHF(2797) = 1                                           CLA05180
                  C TRUNCATION (2 - NO TRUNCATION)                             CLA05190
ISN 0291            IHEDHF(2799) = 2                                           CLA05200
                  C CHANNELS REQUIRED (10000000)                               CLA05210
ISN 0292            IHEDHF(2800) = K200                                        CLA05220
                  C REPEAT OF PIXELS IN SCAN                                   CLA05230
ISN 0293            IHEDHF(2976)=6                                             CLA05240
                  C REPEAT OF SCANS                                            CLA05250
ISN 0294            IHEDHF(2977) = K10                                         CLA05260
                  C                                                            CLA05270
                  C     LOAD USER SUPPLIED DATA INTO HEADER RECORD.  THE DATA IS CLA05280
                  C     THE VERTICES OF THE FIELD USED TO TRAIN THE CLASSIFIER. CLA05290
                  C                                                            CLA05300
ISN 0295            IHEDHF(2941) = KT                                          CLA05310
ISN 0296            IHEDHF(2942) = KR                                          CLA05320
ISN 0297            IHEDHF(2943) = KBLANK                                      CLA05330
ISN 0298            IHEDHF(2944) = KF                                          CLA05340
ISN 0299            IHEDHF(2945) = KL                                          CLA05350
ISN 0300            IHEDHF(2946) = KD                                          CLA05360
ISN 0301            IHEDHF(2947) = KBLANK                                      CLA05370
ISN 0302            IHEDHF(2948) = KFINAL                                      CLA05380
                  C                                                            CLA05390
ISN 0303            REWIND 30                                                  CLA05400
ISN 0304            WRITE (30,4022) (VERTPX(I),VERTLN(I), I = 1,4)             CLA05410
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (APR 78)          MAIN          OS/360  FORTRAN H EXTENDED          DATE 81.139/13.16.48      PAGE    8

 ISN 0305            REWIND 10                                                              CLA05420
 ISN 0306            READ (10,14022) (IHEDHF(I),I = 2949,2996)                              CLA05430
 ISN 0307      14022 FORMAT (48A1)                                                          CLA05440
CCCC  +WITE (3,14022) (IHEDHF(I),I = 2949,2996)                                           CLA05450
 ISN 0308       4022 FORMAT(4(1X,F4.0,1X,F4.0,1X))                                        CLA05460
C                                                                                         CLA05470
C       WRITE HEADER RECORD TO OUTPUT FILE                                                CLA05480
 ISN 0309            WRITE(4,8888) (IHEDHF(I),I = 1,3060)                                  CLA05490
 ISN 0310      8888  FORMAT (15(1X80A1))                                                  CLA05500
C                                                                                         CLA05510
C  ********************************************************************************     CLA05520
C  ********************************************************************************     CLA05530
C  SECTION 4--READS AND PROCESSES THE INDIVIDUAL PIXELS IN THE TEST FLD                   CLA05540
C  USING CONSTANTS CALCULATED IN SECT. 2 (XMNTRN, SGMATR, TRAINA, TRALPH                  CLA05550
C    AND CALLS -TROXM- TO CALCULATE ALPHA (IALPHF), BETA (IBETHF),                        CLA05560
C    TO (ITOHHF) AND SIGMA (ICHIHF) FOR EACH PIXEL IN THE TEST FLD                        CLA05570
C                                                                                         CLA05580
C                                                                                         CLA05590
 ISN 0311            ITYPE = 1                                                             CLA05600
C  READ HEADER AND COORDINATES FOR TEST FLD                                               CLA05610
 ISN 0312            CALL RDFLES (ITYPE, IHEDHF, VERTPX, VERTLN, IERROR)                   CLA05620
CCCC  IF (IERROR .NE. 0) STOP                                                             CLA05630
C                                                                                         CLA05640
C  DETERMIN LINE NUMBER AND PIXEL NUMBERS MAX AND MIN                                     CLA05650
 ISN 0313            LMAX = 1                                                              CLA05660
 ISN 0314            LMIN = 196                                                            CLA05670
 ISN 0315            MINPXL = 196                                                          CLA05680
 ISN 0316            MAXPXL = 1                                                            CLA05690
 ISN 0317            DO 106 KORNER = 1,4                                                   CLA05700
 ISN 0318            LINENO = VERTLN(KORNER)                                               CLA05710
 ISN 0319            IPIXEL = VERTPX(KORNER)                                               CLA05720
 ISN 0320            IF (LINENO .GT. LMAX) LMAX = LINENO                                   CLA05730
 ISN 0322            IF (LINENO .LT. LMIN) LMIN = LINENO                                   CLA05740
 ISN 0324            IF (IPIXEL .GT. MAXPXL) MAXPXL = IPIXEL                               CLA05750
 ISN 0326            IF (IPIXEL .LT. MINPXL) MINPXL = IPIXEL                               CLA05760
 ISN 0328      106   CONTINUE                                                              CLA05770
C                                                                                         CLA05780
C  IPRINT USED TO TRACK VALUE OF I                                                        CLA05790
 ISN 0329            IPRINT = -1                                                           CLA05800
C                                                                                         CLA05810
C   ***********  PROCESS INDIVIDUAL LINES OF TEST FLD ***********                         CLA05820
 ISN 0330            DO 740 KURLNE = LMIN, LMAX                                            CLA05830
 ISN 0331            LINENO = KURLNE                                                       CLA05840
C                                                                                         CLA05850
C   CLEAR THE BUFFER                                                                      CLA05860
 ISN 0332            DO 992 KRR = 1,196                                                    CLA05870
 ISN 0333            IALPHF(KRR) = 0                                                       CLA05880
 ISN 0334            IBETHF(KRR) = 0                                                       CLA05890
 ISN 0335            ICHIHF(KRR) = 0                                                       CLA05900
 ISN 0336      992   ITOHHF(KRR) = 0                                                       CLA05910
C                                                                                         CLA05920
C   READ DATA FOR ONE LINE OF PIXELS.                                                     CLA05930
C                                                                                         CLA05940
 ISN 0337            ITYPE = 2                                                             CLA05950
 ISN 0338            CALL RDFLES (ITYPE, IDATA, VERTPX, VERTLN, IERROR)                    CLA05960
CCCC  IF (IERROR .NE. 0) STOP                                                             CLA05970
C                                                                                         CLA05980
C  MOVE PIXEL DATA FROM UNMERGED FILES                                                    CLA05990
 ISN 0339            DO 412 KURACO = 1,NOACO                                               CLA06000
C                                                                                         CLA06010
 ISN 0340            DO 411 KURCHN = 1,4                                                   CLA06020
 ISN 0341            INCRF = (KURCHN - 1) * 196                                            CLA06030
 ISN 0342            DO 411 KURPXL = MINPXL, MAXPXL                                        CLA06040
C  SKIP OVER FIRST 72 CHARACTERS OF LEAD DATA ON TAPE RECORD                              CLA06050
 ISN 0343            LOC = INCRF + KURPXL + 72                                             CLA06060
 ISN 0344            IHOLD1(4) = IDATA(LOC,KURACO)                                         CLA06070
 ISN 0345            ICHIPX(KURPXL, KURCHN ,KURACO) = IHOLD4                               CLA06080
 ISN 0346      411   CONTINUE                                                              CLA06090
 ISN 0347      412   CONTINUE                                                              CLA06100
C                                                                                         CLA06110
C                                                                                         CLA06120
C     ***********  PROCESS PIXELS FOR CURRENT LINE  ***********                           CLA06130
 ISN 0348            DO 770 KURPXL = MINPXL, MAXPXL                                        CLA06140
 ISN 0349            IPIXEL = KURPXL                                                       CLA06150
C                                                                                         CLA06160
C  CHECK IF PIXEL IN TEST FIELD                                                           CLA06170
 ISN 0350            XLINE = LINENO                                                        CLA06180
 ISN 0351            XPIXEL = IPIXEL                                                       CLA06190
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0352            CALL RECT (XLINE, XPIXEL, VERTLN, VERTPX, ICOR)                    CLA06200
ISN 0353            IF (ICOR .EQ. 1) GO TO 770                                         CLA06210
                    C                                                                  CLA06220
                    C                                                                  CLA06230
                    C   REMOVE THE SCREEN FACTOR FROM  THE PIXEL IF NECESSARY.         CLA06240
ISN 0355            DO 432 II = 1,NOACOS                                               CLA06250
ISN 0356            IF (ICHNPX(IPIXEL,4,II) .GT. 128) ICHNPX(IPIXEL,4,II) =            CLA06260
                   1  ICHNPX(IPIXEL,4,II) - 128                                       CLA06270
ISN 0358        432 CONTINUE                                                           CLA06280
                    C                                                                  CLA06290
ISN 0359            DO 12 M = 1,NOACOS                                                 CLA06300
ISN 0360            INDEX = LANDST(M) - 1                                              CLA06310
ISN 0361            XPIXGR(M) = (  -.29 * ICHNPX(IPIXEL,1,M) * CLNOST(INDEX,1) -      CLA06320
                   1          .562 * ICHNPX(IPIXEL,2,M) * CLNOST(INDEX,2) +        CLA06330
                   2          .6   * ICHNPX(IPIXEL,3,M) * CLNOST(INDEX,3) +        CLA06340
                   3 .2455 * 2. * ICHNPX(IPIXEL,4,M) * CLNOST(INDEX,4) ) * SANG(M)+10.CLA06350
ISN 0362            IF(XPIXGR(M).LE.3.) GO TO 770                                      CLA06360
ISN 0364        12  CONTINUE                                                           CLA06370
                    C                                                                  CLA06380
                    C  MOVE MODEL VALUES CALC'D BY -TRJXM- FIT OF TRAINING FLD PIXELS TO ZX CLA06390
                    C     ARRAY ZX TO BE USED AS INPUT VALUES IN FITTING CURRENT TEST PIXEL CLA06400
ISN 0365            ZX(1) = TRAINA                                                     CLA06410
ISN 0366            ZX(2) = TRALPH                                                     CLA06420
ISN 0367            ZX(3) = TRBETA                                                     CLA06430
ISN 0368            ZX(4) = TRTO                                                       CLA06440
                    C                                                                  CLA06450
ISN 0369            XCHIN = 200.                                                       CLA06460
ISN 0370            XLAMDA = 1.E-03                                                    CLA06470
                    C                                                                  CLA06480
                    C  TRY UNTIL ACCEPTABLE FIT OR MAXIMUM OF 24 TIMES                 CLA06490
ISN 0371            DO 13 LA = 1,LIMIT                                                 CLA06500
ISN 0372            CALL TRJXM(XDAY,XPIXGR,SGMATR,NOACOS,  4,  MODE,ZX,BETA,SIGMA,     CLA06510
                   1 XLAMDA,DFIX,XISQR,0, WEIGHT, IPRINT)                              CLA06520
ISN 0373            XDIF = XCHI-XISQR                                                  CLA06530
ISN 0374            XTH = XISQR*TOL                                                    CLA06540
ISN 0375            IF(ABS(XDIF).LE.XTH) GO TO 14                                      CLA06550
ISN 0377            XCHIN = XISQR                                                      CLA06560
ISN 0378        13  CONTINUE                                                           CLA06570
ISN 0379        14  CONTINUE                                                           CLA06580
                    C                                                                  CLA06590
                    C  IPRINT IS A DEBUG SWITCH TO TRACK A(I), +1--NO PRINT            CLA06600
ISN 0380            IPRINT = +1                                                        CLA06610
                    C                                                                  CLA06620
                    C  CHECK LIMITS AND SAVE VALUES OF ALPHA, BETA, TO AND SIGMA AS INTEGERSCLA06630
                    C   ALPHA                                                         CLA06640
ISN 0381            XALPHA = 100*ZX(2)                                                 CLA06650
ISN 0382            IF(XALPHA.LE.0.0) GO TO 770                                        CLA06660
ISN 0384            IF(XALPHA.GE.9999.) XALPHA = 9999.                                 CLA06670
ISN 0386            IALPHA = XALPHA                                                    CLA06680
                    C                                                                  CLA06690
                    C  BETA                                                            CLA06700
ISN 0387            XBETA = 1000.*ZX(3)                                                CLA06710
ISN 0388            IF(XBETA.LE.0.0) XBETA = 0.0                                       CLA06720
ISN 0390            IF(XBETA.GE.9999.) XBETA = 9999.                                   CLA06730
ISN 0392            IBETA = XBETA                                                      CLA06740
                    C                                                                  CLA06750
                    C  TO                                                              CLA06760
ISN 0393            XTZERO = 1000*ZX(4)                                                CLA06770
ISN 0394            IF(XTZERO.GE.9999.) XTZERO = 9999.                                 CLA06780
ISN 0396            IF(XTZERO.LE.0.0) GO TO 770                                        CLA06790
ISN 0398            ITZERO = XTZERO                                                    CLA06800
                    C                                                                  CLA06810
                    C  SIGMA                                                           CLA06820
ISN 0399            XCHISQ = 10.*XISQR                                                 CLA06830
ISN 0400            IF(XCHISQ.GE.9999.) XCHISQ = 9999.                                 CLA06840
ISN 0402            ICHISQ = XCHISQ                                                    CLA06850
                    C                                                                  CLA06860
                    C  MOVE ALPHA, BETA, TO AND SIGMA TO OUTPUT BUFFER                 CLA06870
ISN 0403            IALPHF(I2IXEL) = IALPHA                                            CLA06880
ISN 0404            IBETAF(I2IXEL) = IBETA                                             CLA06890
ISN 0405            ITOHF(IPIXEL) = ITZERO                                             CLA06900
ISN 0406            ICHIF(IPIXEL) = ICHISQ                                             CLA06910
                                                                                       CLA06920
                    C  CONTINUE PROCESSING PIXELS IN LINE UNTIL END OF PIXEL DO LOOP    CLA06930
ISN 0407        770 CONTINUE                                                           CLA06940
                    C                                                                  CLA06950
                    C                                                                  CLA06960
ISN 0408            WRITE(4,8888)((IALPHF(N),IBETAF(N),ITOHF(N),ICHIF(N)),N=M,NPXL,    CLA06970
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
                  1   MAXPXL)                                                                CLA06980
ISN 0409    8883    FORMAT (1H(45A4))                                                          CLA06990
            CCCC    IF (LINENO.EQ.1) WRITE(6,999) (IALPHF(N), IBETHF(N), ITOHUF(N),          CLA07000
            CCCC  1   ICHIHF(N), N = 1,20)                                                    CLA07010
ISN 0410    999     FORMAT (' ALPHA, BETA, TO, CHI SQ=',4I6)                                  CLA07020
            C                                                                                 CLA07030
            C  CONTINUE PROCESSING LINES, UNTIL END OF LINE DO LOOP                           CLA07040
ISN 0411    780       CONTINUE                                                                CLA07050
            C                                                                                 CLA07060
            C                                                                                 CLA07070
            C  ******************************************                                     CLA07080
            C  CALCULATE AND PRINT ELAPSED TIMES FOR PROGRAM                                  CLA07090
ISN 0412          CALL CPTIME (TODAY, NTIME, NCPTIM,NVCPTM)                                   CLA07100
ISN 0413          TIMECP = (NCPTIM - ICPTIM) / 6000.                                          CLA07110
ISN 0414          TIMEVR = (NVCPTM - IVCPTM) / 6000.                                          CLA07120
ISN 0415          WRITE (6,6010) TIMECP, TIMEVR                                               CLA07130
ISN 0416    6010  FORMAT(///,10X,'CLASSIFICATION FILE -    RUN TIME:  CP=',F6.2,              CLA07140
                1   ' VIRTUAL CP=',F6.2, ' IN MINUTES')                                       CLA07150
            C                                                                                 CLA07160
ISN 0417          IDIFFC = NCPTIM - ICPTIM                                                    CLA07170
ISN 0418          IDIFFV = NVCPTM - IVCPTM                                                    CLA07180
ISN 0419          WRITE (6,6011) IDIFFC, IDIFFV                                               CLA07190
ISN 0420    6011  FORMAT (//,' DIFFERENCE IN INITIAL, ENDING CP TIMES IN HUNDREDTHSCLA07200
                1 SECS.=', I6,/, '  DIFFERENCE IN INITIAL, ENDING VIRTUAL TIMES IN CLA07210
                2HUNDREDTHS SECS.=',I6)                                                       CLA07220
            C                                                                                 CLA07230
            C                                                                                 CLA07240
ISN 0421          STOP                                                                        CLA07250
ISN 0422          END                                                                         CLA07260
```

```
                    *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL  INTERNAL STATEMENT NUMBERS
I       0046 0047 0050 0050 0050 0051 0051 0051 0061 0062 0063 0065 0068 0077 0078 0078 0089 0089 0089
        0089 0160 0160 0160 0162 0163 0167 0173 0174 0174 0174 0192 0193 0218 0219 0221 0222 0261 0262
        0262 0263 0263 0264 0264 0265 0265 0266 0266 0267 0267 0269 0269 0271 0271 0275 0276 0287 0288
        0288 0304 0304 0304 0304 0306 0306 0306 0309 0309 0309
J       0128 0129 0134 0135 0136 0136 0136 0136 0136 0136
K       0151 0152 0152 0186 0186 0186 0186
M       0150 0152 0155 0156 0157 0157 0157 0359 0360 0361 0361 0361 0361 0361 0361 0362
N       0408 0408 0408 0408 0408 0408
II      0355 0356 0355 0356
JM      0081 0082 0082 0082 0095 0096 0097
KA      0025 0029 0210 0281
KB      0025 0029
KC      0025 0028 0208 0274
KD      0025 0026 0300
KE      0025 0028 0203 0216
KF      0025 0028 0214 0284 0299
KG      0025 0028
KH      0025 0029
KI      0025 0029 0213 0215
KJ      0025 0029
KK      0025 0029
KL      0025 0029 0199 0209 0280 0299
KM      0025 0029 0197 0204
KN      0025 0029
KO      0025 0030 0295
KP      0025 0030 0205
KR      0025 0030 0195 0206 0217 0278 0246 0236
KS      0025 0030 0194 0211 0212 0277 0282 0283
KT      0026 0030 0200 0202 0295
KU      0026 0030 0198
K1      0026 0220
LA      0176 0371
ZX      0010 0167 0178 0186 0188 0189 0190 0191 0365 0366 0367 0368 0372 0381 0387 0393
ABS     0181 0375
ANG     0065 0066 0066 0068
III     0072 0072 0072 0074 0074 0074
KHB     0332 0333 0334 0335 0336
K10     0038 0246 0234
K20     0038 0256
LOC     0116 0117 0343 0344
SIN     0068
XTH     0180 0181 0374 0375
BSEG    0023 0024 0239 0240 0241 0242
DATE    0006
ICOR    0125 0126 0352 0353
```



ORIGINAL PAGE IS
OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| SYMBOL | INTERNAL | STATEMENT | NUMBERS | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IDAY | 0023 | 0227 | 0229 | | | | | | | | | | | | | | | |
| IMON | 0023 | 0227 | 0230 | | | | | | | | | | | | | | | |
| KREP | 0080 | 0082 | 0157 | | | | | | | | | | | | | | | |
| K100 | 003B | 0252 | | | | | | | | | | | | | | | | |
| K150 | 003B | 0250 | | | | | | | | | | | | | | | | |
| K200 | 003B | 0241 | 0292 | | | | | | | | | | | | | | | |
| K304 | 003B | 0244 | 0254 | 0258 | | | | | | | | | | | | | | |
| LMAX | 0091 | 0098 | 0098 | 0109 | 0313 | 0320 | 0320 | 0330 | | | | | | | | | | |
| LMIN | 0092 | 0100 | 0100 | 0108 | 0114 | 0322 | 0322 | 0330 | | | | | | | | | | |
| MODE | 0036 | 0178 | 0372 | | | | | | | | | | | | | | | |
| OFIX | 001B | 0178 | 0372 | | | | | | | | | | | | | | | |
| RECT | 0125 | 0352 | | | | | | | | | | | | | | | | |
| SAVG | 0012 | 0062 | 0068 | 0136 | 0361 | | | | | | | | | | | | | |
| SDEV | 0154 | | | | | | | | | | | | | | | | | |
| TRTO | 0191 | 0368 | | | | | | | | | | | | | | | | |
| XDAY | 0017 | 0079 | 0178 | 0372 | | | | | | | | | | | | | | |
| XDIF | 0179 | 0181 | 0373 | 0375 | | | | | | | | | | | | | | |
| XTOL | 003A | 0190 | 0374 | | | | | | | | | | | | | | | |
| ACODT | 0006 | 0007 | 0009 | 0072 | 0160 | | | | | | | | | | | | | |
| CROPG | 0045 | | | | | | | | | | | | | | | | | |
| IBETA | 0392 | 0404 | | | | | | | | | | | | | | | | |
| ICROP | 0010 | 0085 | 0238 | | | | | | | | | | | | | | | |
| IDATA | 0003 | 0004 | 0111 | 0117 | 0338 | 0344 | | | | | | | | | | | | |
| IDATE | 0020 | 0074 | 0078 | 0082 | 0082 | | | | | | | | | | | | | |
| INCRE | 0114 | 0116 | 0341 | 0343 | | | | | | | | | | | | | | |
| INDEX | 0135 | 0136 | 0136 | 0136 | 0136 | 0360 | 0361 | 0361 | 0361 | 0361 | | | | | | | | |
| ITYPE | 0087 | 0088 | 0110 | 0111 | 0311 | 0312 | 0337 | 033A | | | | | | | | | | |
| IYEAR | 0023 | 0227 | 0231 | | | | | | | | | | | | | | | |
| JSFGM | 0024 | 0237 | | | | | | | | | | | | | | | | |
| KSFGM | 0006 | 0035 | 0234 | | | | | | | | | | | | | | | |
| LIMIT | 0037 | 0371 | | | | | | | | | | | | | | | | |
| NMODE | 0171 | 0178 | | | | | | | | | | | | | | | | |
| NOACO | 0006 | 0044 | 0050 | 0051 | 0060 | 0072 | 0074 | 0112 | 0160 | 0339 | | | | | | | | |
| NTERM | 0172 | 017A | 0186 | | | | | | | | | | | | | | | |
| NTIME | 0014 | 0412 | | | | | | | | | | | | | | | | |
| PELTA | 0021 | 0178 | 0372 | | | | | | | | | | | | | | | |
| SIGMA | 0021 | 0178 | 0186 | 0372 | | | | | | | | | | | | | | |
| STIME | 0014 | 0039 | 0085 | 0085 | | | | | | | | | | | | | | |
| TODAY | 0013 | 0039 | 0085 | 0224 | 0412 | | | | | | | | | | | | | |
| TRJXM | 017A | 0372 | | | | | | | | | | | | | | | | |
| XBETA | 0387 | 0398 | 0398 | 0398 | 0398 | 0392 | | | | | | | | | | | | |
| XCHIN | 0170 | 0179 | 0183 | 0369 | 0373 | 0377 | | | | | | | | | | | | |
| XISQR | 017B | 0179 | 0180 | 0183 | 0186 | 0372 | 0373 | 0374 | 0377 | 0399 | | | | | | | | |
| XLINE | 0123 | 0125 | 0350 | 0352 | | | | | | | | | | | | | | |
| XMEAN | 0154 | 0155 | | | | | | | | | | | | | | | | |
| ACODTR | 000B | 0009 | 0262 | 0263 | 0264 | 0265 | 0266 | 0267 | 0269 | 0271 | | | | | | | | |
| CLNDST | 0016 | 0032 | 0032 | 0032 | 0033 | 0034 | 0034 | 0034 | 0035 | 0136 | 0136 | 0136 | 0136 | 0361 | 0361 | 0361 | 0361 | |
| CPTIME | 0039 | 0412 | | | | | | | | | | | | | | | | |
| ERRSET | 0040 | 0041 | 0042 | | | | | | | | | | | | | | | |
| IALPBF | 0022 | 0334 | 0403 | 0408 | | | | | | | | | | | | | | |
| IALPHA | 0386 | 0403 | | | | | | | | | | | | | | | | |
| IBETAF | 0022 | 0334 | 0404 | 0408 | | | | | | | | | | | | | | |
| ICHIBF | 0022 | 0335 | 0406 | 0408 | | | | | | | | | | | | | | |
| ICHISQ | 0402 | 0406 | | | | | | | | | | | | | | | | |
| ICHNPX | 0015 | 0119 | 0129 | 0136 | 0136 | 0136 | 0136 | 0345 | 0356 | 0356 | 0356 | 0361 | 0361 | 0361 | 0361 | | | |
| ICPTIM | 0039 | 0413 | 0417 | | | | | | | | | | | | | | | |
| IDIFFC | 0417 | 0419 | | | | | | | | | | | | | | | | |
| IDIFFV | 041A | 0419 | | | | | | | | | | | | | | | | |
| IERROR | 0045 | 0057 | 0088 | 0111 | 0312 | 033B | | | | | | | | | | | | |
| IHEDBF | 0022 | 0088 | 0193 | 0194 | 0195 | 0196 | 0197 | 0198 | 0199 | 0200 | 0201 | 0202 | 0203 | 0204 | 0205 | 0206 | 0207 | 0208 | 0209 |
| | 0210 | 0211 | 0212 | 0213 | 0214 | 0215 | 0216 | 0217 | 0219 | 0220 | 0222 | 0229 | 0230 | 0231 | 0232 | 0239 | 0240 | 0241 | 0242 |
| | 0243 | 0244 | 0245 | 0246 | 0247 | 0248 | 0249 | 0250 | 0251 | 0252 | 0253 | 0254 | 0255 | 0256 | 0257 | 0258 | 0259 | 0260 | 0262 |
| | 0263 | 0264 | 0265 | 0266 | 0267 | 0269 | 0271 | 0274 | 0276 | 0277 | 027A | 0279 | 0280 | 0281 | 0282 | 0283 | 0284 | 0285 | 0286 |
| | 0288 | 0289 | 0290 | 0291 | 0292 | 0293 | 0294 | 0295 | 0296 | 0297 | 0298 | 0299 | 0300 | 0301 | 0302 | 0306 | 0309 | 0312 | |
| IHOLD1 | 0004 | 0005 | 0117 | 0344 | | | | | | | | | | | | | | | |
| IHOLD4 | 0005 | 0031 | 0118 | 0345 | | | | | | | | | | | | | | | |
| IPIEFL | 0097 | 0102 | 0102 | 0104 | 0104 | 0122 | 0124 | 0129 | 0136 | 0136 | 0136 | 0136 | 0419 | 0324 | 0324 | 0326 | 0326 | 0349 | 0351 |
| | 0356 | 0356 | 0356 | 0361 | 0361 | 0361 | 0361 | 0403 | 0404 | 0405 | 0406 | | | | | | | | |
| IPRINT | 0177 | 017B | 0329 | 0472 | 0480 | | | | | | | | | | | | | | |
| ITZERO | 0398 | 0405 | | | | | | | | | | | | | | | | | |
| ITOBUF | 0022 | 0336 | 0405 | 0408 | | | | | | | | | | | | | | | |
| IVCPTM | 0039 | 0414 | 0418 | | | | | | | | | | | | | | | | |
| KBLANK | 0027 | 0031 | 0196 | 0201 | 0207 | 0219 | 0222 | 0275 | 0297 | 0301 | | | | | | | | | |
| KEQUAL | 0027 | 0302 | | | | | | | | | | | | | | | | | |
| KORNER | 0317 | 0319 | 0319 | | | | | | | | | | | | | | | | |



ORIGINAL PAGE IS
OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

SYMBOL  INTERNAL STATEMENT NUMBERS

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| KURACO | 0112 | 0117 | 0118 | 0339 | 0344 | 0345 | | | | | | | | | |
| KURCHN | 0113 | 0114 | 0118 | 0340 | 0341 | 0345 | | | | | | | | | |
| KURLNF | 0108 | 0109 | 0330 | 0331 | | | | | | | | | | | |
| KURPXL | 0115 | 0116 | 0118 | 0121 | 0122 | 0342 | 0343 | 0345 | 0348 | 0349 | | | | | |
| LANDST | 0006 | 0135 | 0360 | | | | | | | | | | | | |
| LINENO | 0003 | 0096 | 0098 | 0098 | 0100 | 0100 | 0109 | 0123 | 0318 | 0320 | 0320 | 0322 | 0322 | 0331 | 0350 |
| MAXPXL | 0096 | 0102 | 0102 | 0115 | 0121 | 0316 | 0324 | 0324 | 0342 | 0348 | 0404 | | | | |
| MINPXL | 0093 | 0104 | 0104 | 0115 | 0121 | 0315 | 0326 | 0326 | 0342 | 0348 | 0404 | | | | |
| NCPTIM | 0412 | 0413 | 0417 | | | | | | | | | | | | |
| NOACOS | 0003 | 0050 | 0051 | 0128 | 0134 | 0150 | 0173 | 0178 | 0267 | 0269 | 0271 | 0355 | 0359 | 0372 | |
| NOTPPX | 0107 | 0132 | 0132 | 0136 | 0139 | 0143 | 0145 | 0151 | 0154 | 0160 | | | | | |
| NVCPTM | 0412 | 0414 | 0414 | | | | | | | | | | | | |
| NXTLNF | 0003 | | | | | | | | | | | | | | |
| PDFXFC | 0043 | | | | | | | | | | | | | | |
| PDFLFS | 0088 | 0111 | 0312 | 0338 | | | | | | | | | | | |
| SAMGGG | 0010 | 0011 | 0047 | 0047 | 0050 | 0051 | 0063 | 0065 | | | | | | | |
| SGMATR | 0020 | 0156 | 0157 | 0157 | 0161 | 0174 | 0174 | 0178 | 0372 | | | | | | |
| SIGNAG | 0154 | 0154 | | | | | | | | | | | | | |
| TIMECP | 0413 | 0415 | | | | | | | | | | | | | |
| TIMEVR | 0414 | 0415 | | | | | | | | | | | | | |
| TRAINA | 0188 | 0365 | | | | | | | | | | | | | |
| TRAIPH | 0189 | 0366 | | | | | | | | | | | | | |
| THRETA | 0190 | 0367 | | | | | | | | | | | | | |
| VERTLN | 0019 | 0088 | 0089 | 0096 | 0111 | 0125 | 0304 | 0312 | 0318 | 0338 | 0352 | | | | |
| VERTPX | 0018 | 0098 | 0099 | 0097 | 0111 | 0125 | 0304 | 0312 | 0319 | 0338 | 0352 | | | | |
| WEIGHT | 0013 | 0174 | 0178 | 0372 | | | | | | | | | | | |
| XALPHA | 0381 | 0382 | 0384 | 0384 | 0386 | | | | | | | | | | |
| XCHISQ | 0398 | 0400 | 0400 | 0402 | | | | | | | | | | | |
| XGREEN | 0019 | 0136 | 0152 | | | | | | | | | | | | |
| XLAMDA | 0169 | 0174 | 0370 | 0372 | | | | | | | | | | | |
| XPHITN | 0021 | 0155 | 0163 | 0178 | | | | | | | | | | | |
| XPIXEL | 0124 | 0125 | 0351 | 0352 | | | | | | | | | | | |
| XPIXGR | 0021 | 0361 | 0362 | 0372 | | | | | | | | | | | |
| XTZERO | 0393 | 0394 | 0394 | 0396 | 0398 | | | | | | | | | | |
| YGREEN | 0020 | 0152 | 0154 | | | | | | | | | | | | |

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES | | | | |
|---|---|---|---|---|---|---|
| 1 | 0084 | 0091 | | | | |
| 3 | 0060 | 0061 | 0063 | | | |
| 5 | 0079 | 0077 | | | | |
| 6 | 0060 | 0046 | | | | |
| 7 | 0054 | 0050 | 0051 | | | |
| 8 | 0057 | 0047 | | | | |
| 9 | 0055 | 0052 | 0053 | | | |
| 12 | 0364 | 0359 | | | | |
| 13 | 0378 | 0371 | | | | |
| 14 | 0379 | 0375 | | | | |
| 17 | 0075 | 0072 | | | | |
| 18 | 0076 | 0074 | | | | |
| 24 | 0106 | 0095 | | | | |
| 26 | 0131 | 0128 | | | | |
| 27 | 0138 | 0134 | | | | |
| 28 | 0147 | 0139 | | | | |
| 29 | 0159 | 0150 | | | | |
| 31 | 0153 | 0151 | | | | |
| 48 | 0150 | 0143 | | | | |
| 49 | 0119 | 0113 | 0115 | | | |
| 50 | 0120 | 0112 | | | | |
| 77 | 0141 | 0121 | 0126 | 0129 | | |
| 78 | 0142 | 0108 | | | | |
| 104 | 0193 | 0192 | | | | |
| 106 | 0324 | 0317 | | | | |
| 114 | 0219 | 0218 | | | | |
| 115 | 0222 | 0221 | | | | |
| 116 | 0273 | 0261 | | | | |
| 117 | 0276 | 0275 | | | | |
| 118 | 0288 | 0287 | | | | |
| 190 | 0007 | | | | | |
| 411 | 0346 | 0340 | 0342 | | | |
| 412 | 0347 | 0339 | | | | |
| 432 | 0358 | 0355 | | | | |
| 770 | 0407 | 0348 | 0353 | 0362 | 0382 | 0395 |
| 780 | 0411 | 0130 | | | | |



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

***** F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G *****

| LABEL | DEFINED | REFERENCES |
|---|---|---|
| 950 | 0146 | 0145 |
| 992 | 0334 | 0332 |
| 999 | 0410 | |
| 1114 | 0225 | 0224 |
| 1115 | 0228 | 0227 |
| 1117 | 0238 | 0237 |
| 1124 | 0235 | 0234 |
| 2999 | 0174 | 0173 |
| 3001 | 0184 | 0176 |
| 3002 | 0185 | 0181 |
| 4022 | 0308 | 0304 |
| 6000 | 0086 | 0085 |
| 6002 | 0090 | 0089 |
| 6003 | 0161 | 0160 |
| 6004 | 0164 | 0163 |
| 6005 | 0166 | 0165 |
| 6006 | 0168 | 0167 |
| 6007 | 0187 | 0186 |
| 6010 | 0415 | 0415 |
| 6011 | 0420 | 0419 |
| 8888 | 0310 | 0309 |
| 8889 | 0409 | 0404 |
| 9001 | 0148 | 0147 |
| 9994 | 0175 | |
| 9995 | 0137 | |
| 9996 | 0133 | |
| 9997 | 0070 | |
| 9998 | 0059 | |
| 9999 | 0044 | |
| 14022 | 0307 | 0305 |

/ MAIN / SIZE OF PROGRAM 009F64 HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I | SF | I*4 | 000940 | J | SF | I*4 | 000950 | K | SF | I*4 | 000954 | M | SF | I*4 | 000958 |
| N | SF | I*4 | 000950 | II | SF | I*4 | 000960 | JM | SF | I*4 | 000964 | KA | F | L*1 | 000930 |
| KB | | L*1 | 000931 | KC | F | L*1 | 000932 | KD | F | L*1 | 000933 | KE | F | L*1 | 000934 |
| KF | F | L*1 | 000935 | KG | | L*1 | 000936 | KH | | L*1 | 000937 | KI | F | L*1 | 000938 |
| KJ | | L*1 | 000939 | KK | | L*1 | 00093A | KL | F | L*1 | 00093B | KM | F | L*1 | 00093C |
| KN | | L*1 | 000930 | KO | F | L*1 | 00093E | KP | F | L*1 | 00093F | KR | F | L*1 | 000940 |
| KS | F | L*1 | 000941 | KT | F | L*1 | 000942 | KU | F | L*1 | 000943 | KL | F | L*1 | 000944 |
| LA | SF | I*4 | 000968 | ZX | SFA C | R*4 | 000010 | ANG | SFA | R*4 | 00096C | III | F | I*4 | 000970 |
| KHH | SF | I*4 | 000974 | K10 | F | I*4 | 000978 | K20 | F | I*4 | 00097C | LOC | SF | I*4 | 000980 |
| SIN | F XF | R*4 | 000000 | XTH | S | R*4 | 000984 | HSFG | F E | L*1 | 000980 | DATE | C | I*4 | NR |
| ICOR | SFA | I*4 | 000988 | IDAY | SF | L*1 | 000945 | IMON | SF | L*1 | 000946 | KREP | S | I*4 | 00098C |
| K106 | F | I*4 | 000990 | K150 | F | I*4 | 000994 | K200 | F | I*4 | 000998 | K304 | F | I*4 | 00099C |
| LMAX | SF | I*4 | 0009A0 | LMIN | SF | I*4 | 0009A4 | MODE | SFA | I*4 | 0009A8 | QFIX | SFA | R*4 | 001674 |
| QFCT | SF XF | | 000000 | SANG | SF | R*4 | 001694 | SDEV | SF XF | | 000000 | IRTO | SF | R*4 | 0009AC |
| XDAY | SFA | R*4 | 001684 | XDIF | SFA | R*4 | 0009B0 | XTOL | F | R*4 | 0009B4 | ACODT | F CE | R*8 | 000008 |
| CROPG | SF XF | | 000000 | IBETA | SF | I*4 | 0009B8 | ICROP | F C | I*4 | 000000 | IDATA | SFA C | L*1 | 000008 |
| IDATE | SF | I*4 | 001604 | INCRE | SF | I*4 | 0009BC | INDEX | SF | I*4 | 0009C0 | ITYPE | SFA | I*4 | 0009C4 |
| IYEAR | SF | L*1 | 000947 | JSEGM | S E | I*4 | 00B780 | KSEG4 | F C | I*4 | 000048 | LIMIT | F | I*4 | 0009C8 |
| NMODF | SFA | I*4 | 0009CC | NOACO | I C | I*4 | 000008 | NTERM | SFA | I*4 | 0009D0 | NTIME | SFA | I*2 | 001664 |
| PELTA | SFA | R*4 | 001684 | SIGMA | SFA | R*4 | 001704 | STIME | SFA | | | TODAY | SFA | R*8 | 000A38 |
| TOJXM | SF XF | | 000000 | XBETA | SF | R*4 | 0009D4 | XCHIN | SF | R*4 | 0009D8 | XISQR | SFA | R*4 | 0009DC |
| XLINF | SFA | R*4 | 6009E0 | XMEAN | SFA | R*4 | 0009E4 | ACODTH | F CE | L*1 | 000008 | CLNDST | F | R*4 | 001714 |
| CPTIMF | SF XF | | 000000 | FORSET | SF XF | | 000000 | IALPRF | SF | I*4 | 001734 | IALPHA | SF | I*4 | 0009E8 |
| IPCONF | F XF | I*4 | 000000 | IRETHF | SF | I*4 | 001A44 | ICHIRF | SF | I*4 | 001054 | ICHISQ | SF | I*4 | 0009EC |
| ICHNPX | SF | I*4 | 002SA8 | ICPTIM | SFA | I*4 | 0009F0 | IDIFFC | SF | I*4 | 0009F4 | IDIFFV | SF | I*4 | 0009F8 |
| IERROR | SFA | I*4 | 0009FC | IHEDHF | SFA | L*1 | 000A70 | IHOLDI | S E | L*1 | 0087A8 | IHOLD4 | F E | I*4 | 0087A8 |
| IPIXEL | SF | I*4 | 000A00 | IPRINT | SFA | I*4 | 000A04 | ITZERO | SF | I*4 | 000A08 | ITOALF | SF | I*4 | 002064 |
| IVCPTM | SFA | I*4 | 000A0C | KHLAIM | F | L*1 | 000942 | KEQUAL | F | L*1 | 000949 | KORNER | SF | I*4 | 000A10 |
| KURACO | SF | I*4 | 000A14 | KURCHI | SF | I*4 | 000A18 | KURLNE | SF | I*4 | 000A1C | KURPXL | SF | I*4 | 000A20 |
| LANDST | F C | I*4 | 000A5C | LINEND | SF C | I*4 | 000A34 | MAXPXL | SF | I*4 | 000A24 | MINPXL | SF | I*4 | 000A28 |
| NCPTIM | SFA C | I*4 | 000A2C | NOACOS | SFA C | I*4 | 001ACH | NOTHPX | SFA | I*4 | 000A30 | NVCPTM | SFA | I*4 | 000A34 |
| NXTLNF | C | I*4 | NU | ROFXEC | SF XF | | 000000 | RULES | SF XF | | 000000 | SANGGG | F C | I*4 | 000000 |
| SGMATR | SFA | R*4 | 002374 | SIGMAG | SFA | R*4 | 000A38 | TIMECH | SF | R*4 | 000A3C | TIMEVR | SF | R*4 | 000A40 |
| TRAINA | SF | R*4 | 000A44 | TRALPH | SF | R*4 | 000A48 | TWBETA | SF | R*4 | 000A4C | VERTLN | SFA | R*4 | 002394 |
| VERTPX | SFA | R*4 | 002384 | WEIGHT | SFA | R*4 | 002568 | XALPHA | SF | R*4 | 000A50 | XCHISQ | SF | R*4 | 000A54 |
| XGREEN | SF | R*4 | 002384 | XLAMDA | SFA | R*4 | 000A58 | XMNTHN | SFA | R*4 | 0024F4 | XPIXEL | SFA | R*4 | 000A5C |
| XPIXGN | SFA | R*4 | 002514 | XTZERO | SF | R*4 | 000A60 | YGREEN | SFA | R*4 | 00253C | | | | |

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * LINES* SIZE OF BLOCK 000ACC HEXADECIMAL BYTES

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MAXLINE | I*4 | 000000 NR | LINENO | I*4 | 000004 | IDATA | I*1 | 000008 | NOACOS | I*4 | 000AC8 |

NAME OF COMMON BLOCK * MISC* SIZE OF BLOCK 00007C HEXADECIMAL BYTES

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NOACO | I*4 | 000000 | ACODT | R*8 | 000008 | KSEGM | I*4 | 000048 | DATE | I*4 | 00004C NR |
| LANDST | I*4 | 000050 | | | | | | | | | |

EQUIVALENCED VARIABLES WITHIN THIS COMMON BLOCK

| VARIABLE | OFFSET | VARIABLE | OFFSET | VARIABLE | OFFSET | VARIABLE | OFFSET |
|---|---|---|---|---|---|---|---|
| ACODTH | 000008 | | | | | | |

NAME OF COMMON BLOCK * USER* SIZE OF BLOCK 000040 HEXADECIMAL BYTES

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| IGROP | I*4 | 000000 | ?x | R*4 | 000010 | SANGGG | I*4 | 000020 | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 49 | 008BAE | 8 | 57 | 008C5E | 3 | 69 | 008A04 | 5 | 79 | 008AC8 |
| 1 | 84 | 008B10 | 199 | 87 | 008B6C NR | 24 | 106 | 008C80 | 49 | 119 | 008D3E |
| 50 | 120 | 008D66 | 26 | 131 | 008E00 | 27 | 138 | 008F48 | 77 | 141 | 008F6C |
| 78 | 142 | 008F80 | 28 | 143 | 008F94 | 48 | 150 | 008FDE | 31 | 153 | 009010 |
| 29 | 159 | 009076 | 2999 | 174 | 00916C | 3001 | 184 | 0091F8 | 3002 | 185 | 00920C |
| 104 | 193 | 0092BC | 114 | 219 | 009374 | 115 | 222 | 0093A4 | 116 | 273 | 0095E8 |
| 117 | 276 | 009600 | 118 | 288 | 009644 | 106 | 328 | 0098C2 | 992 | 336 | 00992C |
| 411 | 346 | 0099DE | 412 | 347 | 009A06 | 432 | 358 | 009AD4 | 12 | 364 | 009C1A |
| 13 | 378 | 009C9C | 14 | 379 | 009C30 | 770 | 407 | 009D0E | 748 | 411 | 009E64 |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100000 | 1 | 008910 | 100001 | 47 | 00894A | 100002 | 50 | 0089C6 | 100007 | 58 | 008968 |
| 100008 | 60 | 008976 | 100009 | 62 | 0089BE | 100010 | 65 | 008A30 | 100011 | 67 | 008A9C |
| 100012 | 68 | 008AE8 | 100013 | 70 | 008A18 | 100014 | 78 | 008A9C | 100019 | 80 | 008ADC |
| 100020 | 82 | 008AEC | 100021 | 83 | 008B08 | 100022 | 85 | 008B28 | 100023 | 86 | 008B9R |
| 100024 | 90 | 008BCC | 100025 | 96 | 008BF0 | 100026 | 99 | 008C38 | 100027 | 100 | 008C3C |
| 100028 | 101 | 008C4C | 100029 | 102 | 008C58 | 100030 | 103 | 008C64 | 100031 | 104 | 008C6C |
| 100032 | 105 | 008C78 | 100033 | 107 | 008C94 | 100034 | 109 | 008CA4 | 100035 | 113 | 008CCA |
| 100036 | 116 | 008CD2 | 100037 | 116 | 008CEA | 100038 | 120 | 008D52 | 100040 | 121 | 008D7E |
| 100041 | 122 | 008D86 | 100042 | 128 | 008DD6 | 100043 | 129 | 008DDE | 100044 | 132 | 008E18 |
| 100045 | 135 | 008E2C | 100046 | 139 | 008E60 | 100049 | 145 | 008FA0 | 100050 | 151 | 008FF6 |
| 100051 | 152 | 008FEF | 100052 | 154 | 009024 | 100053 | 154 | 009054 | 100054 | 160 | 009088 |
| 100057 | 175 | 0091B4 | 100058 | 177 | 0091HC | 100059 | 183 | 0091F0 | 100061 | 186 | 009224 |
| 100062 | 186 | 009258 | 100063 | 194 | 0092AC | 100064 | 220 | 009394 | 100065 | 223 | 0093C4 |
| 100066 | 249 | 0094F0 | 100066 | 262 | 009558 | 100067 | 268 | 009598 | 100068 | 269 | 0095A8 |
| 100069 | 270 | 0095H8 | 100070 | 271 | 0095C8 | 100071 | 272 | 0095D8 | 100072 | 274 | 0095FC |
| 100073 | 277 | 009620 | 100074 | 289 | 0096AC | 100075 | 304 | 009768 | 100076 | 304 | 00979C |
| 100081 | 318 | 009832 | 100082 | 321 | 009A7A | 100083 | 322 | 009B7F | 100084 | 323 | 009B8E |
| 100085 | 324 | 009B9A | 100086 | 325 | 009BA6 | 100087 | 326 | 009BAE | 100088 | 327 | 009BBA |
| 100089 | 329 | 009BD6 | 100090 | 331 | 009BEB | 100091 | 333 | 009BFC | 100092 | 337 | 009950 |
| 100093 | 340 | 009976A | 100094 | 341 | 009972 | 100095 | 343 | 009998 | 100096 | 347 | 0099F2 |
| 100098 | 348 | 009A1E | 100099 | 349 | 009A26 | 100100 | 355 | 009A76 | 100101 | 356 | 009A7E |
| 100102 | 357 | 009AA0 | 100103 | 359 | 009AEC | 100104 | 360 | 009AF4 | 100105 | 365 | 009C32 |
| 100106 | 372 | 009C6A | 100107 | 377 | 009C94 | 100109 | 344 | 009CCA | 100110 | 385 | 009CD2 |
| 100111 | 386 | 009CDA | 100112 | 389 | 009D02 | 100113 | 390 | 009D0A | 100114 | 391 | 009D16 |
| 100115 | 392 | 009D1E | 100116 | 395 | 009D48 | 100117 | 396 | 009D50 | 100118 | 398 | 009D5A |
| 100119 | 401 | 009DA4 | 100120 | 402 | 009D4C | 100121 | 404 | 009DF2 | 100122 | 408 | 009E0C |
| 100123 | 409 | 009E60 | 100124 | 412 | 009E78 | | | | | | |

FORMAT STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9999 | 44 | 000028 NR | 7 | 54 | 000065 | 9 | 55 | 000088 | 9998 | 59 | 0000B8 NR |
| 9997 | 78 | 000005 NR | 17 | 75 | 0000EF | 18 | 76 | 000104 | 6000 | 86 | 00010F |
| 6002 | 90 | 000186 | 9996 | 133 | 0001E5 NR | 9995 | 137 | 0001FF NR | 950 | 146 | 000223 |
| 9001 | 148 | 000288 | 6003 | 161 | 0002AD | 6004 | 164 | 00033F | 6005 | 166 | 00036A |
| 6006 | 168 | 0003C5 | 9994 | 175 | 000B73 NR | 6007 | 187 | 000420 | 1114 | 225 | 000444 |
| 1115 | 228 | 000448 | 1124 | 235 | 000454 | 1117 | 238 | 000458 | 14022 | 307 | 0004SC |



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)          MAIN          OS/360  FORTRAN H EXTENDED          DATE 81.139/13.16.49          PAGE  15
     4022  309  000462          8088  310  000477          8989  409  000480          949  410  000489 NR
     6010  415  0004AA          6011  426  0004FA
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EXPDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =      421, PROGRAM SIZE =     40804, SUBPROGRAM NAME =  MAIN
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                  212K BYTES OF CORE NOT USED
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (APR 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.139/13.20.22       PAGE   1

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002            SUBROUTINE CROPG (IERROR)                                                    CR000010
                  C                                                                                 CR000020
   ISN 0003            IMPLICIT INTEGER (A-Z)                                                       CR000030
                  C                                                                                 CR000040
                  C  CROPGF:  READ AND ANALYZE LINES DESCRIBING THE FOLLOWING--                     CR000050
                  C    CROP      CROP TO BE EVALUATED  (4A1)                                        CR000060
                  C    INITIAL VALUES FOR 4 CHANNELS                                              CR000070
                  C                                                                                 CR000080
                  C                                                                                 CR000090
   ISN 0004            COMMON /USER/ ICROP(4),ZX(4),SUNANG(8)                                       CR000100
   ISN 0005            REAL*4 ZX                                                                    CR000110
   ISN 0006            DIMENSION LINE(80), ITEMP(64)                                                CR000120
                  C                                                                                 CR000130
   ISN 0007            DIMENSION ITYPES(5)                                                          CR000140
                  C                    1       2       3       4                                    CR000150
   ISN 0008             DATA ITYPES /'CROP', 'GWEE', 'COMM','SUNA','END'/                           CR000160
                  C                                                                                 CR000170
   ISN 0009            DATA IBLANK/' '/                                                             CR000180
                  C                                                                                 CR000190
                  C     ***********  INITIALIZE VARIABLES SET FROM CROP FILE   ***********          CR000200
                  C                                                                                 CR000210
   ISN 0010            IERROR = 0                                                                   CR000220
                  C                                                                                 CR000230
   ISN 0011            DO 3 I = 1,4                                                                 CR000240
   ISN 0012            ICROP(I) = IBLANK                                                            CR000250
   ISN 0013            ZX(I) = 0.                                                                   CR000260
   ISN 0014          3 CONTINUE                                                                     CR000270
                  C                                                                                 CR000280
                  C                                                                                 CR000290
                  C  ********  WRITE HEADING FOR CROP INFORMATION SUMMARY  ***********              CR000300
   ISN 0015            WRITE(6,610)                                                                 CR000310
   ISN 0016         610 FORMAT(' CROP INFORMATION SUMMARY',//,' TYPE  DATA',/)                       CR000320
                  C                                                                                 CR000330
                  C        ***********  READ LINE, PRINT LINE ON SUMMARY  ********                  CR000340
                  C  READ LINE                                                                      CR000350
   ISN 0017         4    READ (21,1000) KRDTYP, (LINE(II), II=11,80)                                 CR000360
   ISN 0018         1000 FORMAT (A4, 6X, 70A1)                                                       CR000370
                  C                                                                                 CR000380
                  C  WRITE INPUT LINE ON SUMMARY REPORT                                             CR000390
   ISN 0019            WRITE (6,1001) KRDTYP, (LINE(II),II=11,80)                                   CR000400
   ISN 0020         1001  FORMAT (1X, A4,2X,70A1)                                                   CR000410
                  C                                                                                 CR000420
                  C   ************  ANALYZE LINE  **************                                    CR000430
                  C  DETERMINE LINE TYPE                                                            CR000440
   ISN 0021            DO 5 I=1,5                                                                   CR000450
   ISN 0022            IF (KRDTYP .EQ. ITYPES(I)) GO TO (10, 20, 4, 40, 50), I                      CR000460
   ISN 0024          5 CONTINUE                                                                     CR000470
                  C                                                                                 CR000480
                  C  INVALID LINE TYPE                                                              CR000490
   ISN 0025            WRITE(6,630)                                                                 CR000500
   ISN 0026         630 FORMAT(' INVALID INPUT LINE----PROGRAM WILL ABORT AFTER READING AL CR000510
                       IL USER DATA',/)                                                             CR000520
   ISN 0027            IERROR = 1                                                                   CR000530
   ISN 0028            GO TO 4                                                                      CR000540
                  C                                                                                 CR000550
                  C   ************  VALID LINE TYPES, PROCESS DATA  ***********                     CR000560
                  C  CROP LINE                                                                      CR000570
   ISN 0029         10   INDXCOP = 1                                                                CR000580
   ISN 0030              INDEX = 10                                                                  CR000590
   ISN 0031         11   INDEX = INDEX + 1                                                          CR000600
   ISN 0032              IF (INDEX .GT. 80) GO TO 4                                                  CR000610
   ISN 0034              IF (LINE(INDEX) .EQ. IBLANK  .OR. LINE(INDEX).EQ.KCOMMA) GO TO 11 CR000620
   ISN 0036              ICROP(INDXCOP) = LINE(INDEX)                                                CR000630
   ISN 0037              INDXCOP = INDXCOP + 1                                                       CR000640
   ISN 0038              IF (INDXCOP .GE. 5) GO TO 4                                                 CR000650
   ISN 0040              GO TO 11                                                                    CR000660
                  C                                                                                 CR000670
                  C  INITIAL GREENESS VALUES LINE                                                   CR000680
   ISN 0041         20   NOTAT = INPUTL(LINE,ZX)                                                     CR000690
   ISN 0042              IF (NOTAT .NE. 4) GO TO 125                                                 CR000700
   ISN 0044              GO TO 4                                                                     CR000710
                  C  ERROR ON INITIAL GREEN LINE                                                    CR000720
   ISN 0045         125  WRITE (6,126)                                                              CR000730
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
   ISN 0046      125  FORMAT (' ERROR ON INITIAL GREENNESS LINE. MUST BE 4 INITIAL QUES  CR000740
                     1SES FOR EACH SET OF GREENNESS VALUES', /, ' PROGRAM WILL STOP')    CR000750
   ISN 0047          STOP                                                               CR000760
                C                                                                      CR000770
                C                                                                      CR000780
                C  SUNANGLE CARD                                                       CR000790
   ISN 0048     40   CALL NEWUP (LINE,SUNANG)                                          CR000800
   ISN 0049          GO TO 4                                                          CR000810
                C                                                                      CR000820
                C  *ERROR LINE                                                         CR000830
   ISN 0050     50   CONTINUE                                                         CR000840
                C                                                                      CR000850
   ISN 0051          RETURN                                                           CR000860
   ISN 0052          END                                                              CR000870
```

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| SYMBOL | INTERNAL | STATEMENT NUMBERS | | | | | |
|---|---|---|---|---|---|---|---|
| I | 0011 | 0012 | 0013 | 0021 | 0022 | 0022 | |
| II | 0017 | 0017 | 0017 | 0019 | 0019 | | |
| IX | 0004 | 0005 | 0013 | 0041 | | | |
| LINE | 0006 | 0017 | 0019 | 0034 | 0034 | 0036 | 0041 | 0048 |
| CROPG | 0002 | | | | | | | |
| ICROP | 0004 | 0012 | 0036 | | | | | |
| INDEX | 0030 | 0031 | 0031 | 0032 | 0034 | 0034 | 0036 | |
| ITEMP | 0006 | | | | | | | |
| NOINT | 0041 | 0042 | | | | | | |
| NUMBR | 0048 | | | | | | | |
| IBLANK | 0000 | 0012 | 0036 | | | | | |
| IERROR | 0002 | 0019 | 0027 | | | | | |
| ITYPES | 0007 | 0008 | 0022 | | | | | |
| KCOMMA | 0034 | | | | | | | |
| KRDTYP | 0017 | 0019 | 0022 | | | | | |
| NDXCRP | 0029 | 0036 | 0037 | 0037 | 0038 | | | |
| NUMHFL | 0041 | | | | | | | |
| SUNANG | 0004 | 0048 | | | | | | |

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES | | | | | |
|---|---|---|---|---|---|---|---|
| 3 | 0014 | 0011 | | | | | |
| 4 | 0017 | 0022 | 0028 | 0032 | 0039 | 0044 | 0049 |
| 5 | 0024 | 0021 | | | | | |
| 10 | 0029 | 0022 | | | | | |
| 11 | 0031 | 0034 | 0040 | | | | |
| 20 | 0041 | 0022 | | | | | |
| 46 | 0044 | 0022 | | | | | |
| 50 | 0050 | 0022 | | | | | |
| 125 | 0045 | 0042 | | | | | |
| 126 | 0046 | 0045 | | | | | |
| 490 | 0026 | 0025 | | | | | |
| 630 | 0016 | 0015 | | | | | |
| 1000 | 0018 | 0017 | | | | | |
| 1001 | 0020 | 0019 | | | | | |

/ CROPG / SIZE OF PROGRAM 000570 HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I | SF | I*4 | 000100 | II | F | I*4 | 000104 | IX | SFA C | I*4 | 000010 | LINE | SFA | I*4 | 0001F8 |
| CROPG | S | I*4 | 000108 | ICROP | S C | I*4 | 000000 | INDEX | SF | I*4 | 00010C | ITEMP | | I*4 | NR |
| NOINT | S | I*4 | 0001E0 | NUMBR | SF XF | | 000000 | KCOMMA | F XF | I*4 | 000000 | IBLANK | F | I*4 | 0001E4 |
| IERROR | S | I*4 | 0001E8 | ITYPES | | I*4 | 000334 | KCOMMA | | I*4 | 0001EC | KRDTYP | SF | I*4 | 0001F0 |
| NDXCRP | SF | I*4 | 0001F4 | NUMHFL | F XF | I*4 | 000000 | SUNANG | SFA C | I*4 | 000020 | | | | |

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK = #BLNK#  SIZE OF BLOCK 000040 HEXADECIMAL BYTES

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ICROP | I*4 | 000000 | IX | I*4 | 000010 | SUNANG | I*4 | 000020 | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|



ORIGINAL PAGE IS
OF POOR QUALITY

```
     3   14  000394          4   17  000BCA          5   24  00045A         10   29  000486
    11   31  0004AC         20   41  0004DC        125   45  0004F2         40   48  000516
```

COMPILER GENERATED LABELS

```
LABEL  ISN  ADDR     LABEL  ISN  ADDR     LABEL  ISN  ADDR     LABEL  ISN  ADDR
100001   2  000360   100002  12  000394   100003  15  000384   100008  22  00042C
100009  24  000440   100011  25  00046A   100012  34  000496   200001  35  0004A8
100013  36  0004B4   100014  40  0004DA   100015  44  0004FE
```

FORMAT STATEMENT LABELS

```
LABEL  ISN  ADDR     LABEL  ISN  ADDR     LABEL  ISN  ADDR     LABEL  ISN  ADDR
  630  14  000028     1000  18  000054     1001  20  00005E      400  26  00006A
  126  46  000044
```

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EXPDCK NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IHM FLAG(I)

*STATISTICS* SOURCE STATEMENTS = 51, PROGRAM SIZE = 1342, SUBPROGRAM NAME = CROPG

*STATISTICS* NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******

292K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                  OS/360  FORTRAN H EXTENDED          DATE 81.141/14.02.20      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002              FUNCTION MOVSYM (CARD, ITEMP)                                          NUM00650
                     C                                                                          NUM00660
                     C  PURPOSE:  DECODE CARD OF FORM   SYMBOL1, SYMBOL2, ... SYMBOLN           NUM00670
                     C                                                                          NUM00680
   ISN 0003              REAL*8 ITEMP(30), IHOLD8                                               NUM00690
   ISN 0004              LOGICAL*1 ICHAR(1), IHOLD(8), IBLNK1                                   NUM00700
   ISN 0005              INTEGER*4 CARD(1)                                                      NUM00710
   ISN 0006              EQUIVALENCE (ICHAR(1), ICHAR4), (IHOLD(1),IHOLD8)                      NUM00720
                     C                                                                          NUM00730
   ISN 0007              DATA IBLNK1/' '/, ICHAR4/' '/, IBLANK/' '/, ICOMMA/','/                NUM00740
                     C                                                                          NUM00750
   ISN 0008              DO 2 I = 1,8                                                           NUM00760
   ISN 0009            2 IHOLD(I) = IBLNK1                                                      NUM00770
                     C                                                                          NUM00780
   ISN 0010              NDEX = 0                                                               NUM00790
   ISN 0011              KOUNT = 0                                                              NUM00800
                     C                                                                          NUM00810
   ISN 0012              DO 100 KOL = 11,80                                                     NUM00820
   ISN 0013              ICHAR4 = CARD(KOL)                                                     NUM00830
   ISN 0014              IF (ICHAR4 .NE. IBLANK) GO TO 10                                       NUM00840
                     C                                                                          NUM00850
                     C  BLANK                                                                   NUM00860
   ISN 0016              IF (KOL .EQ. 80 .AND. NDEX .NE. 0) GO TO 80                            NUM00870
   ISN 0018              GO TO 100                                                              NUM00880
                     C                                                                          NUM00890
                     C  NOT BLANK                                                               NUM00900
   ISN 0019           10 IF (ICHAR4 .EQ. ICOMMA) GO TO 80                                       NUM00910
                     C                                                                          NUM00920
                     C  STORE CHARACTER IN WORD DESCRIBING SYMBOL, MAX OF 4 CHARACTERS          NUM00930
   ISN 0021              NDEX = NDEX + 1                                                        NUM00940
   ISN 0022              IF (NDEX .GT. 8) GO TO 100                                             NUM00950
   ISN 0024              IHOLD(NDEX) = ICHAR(I)                                                 NUM00960
   ISN 0025              IF (KOL .EQ. 80) GO TO 80                                              NUM00970
   ISN 0027              GO TO 100                                                              NUM00980
                     C                                                                          NUM00990
                     C  END OF SYMBOL                                                           NUM01000
   ISN 0028           80 KOUNT = KOUNT + 1                                                      NUM01010
   ISN 0029              ITEMP(KOUNT) = IHOLD8                                                  NUM01020
   ISN 0030              NDEX = 0                                                               NUM01030
   ISN 0031              DO 81 II = 1,8                                                         NUM01040
   ISN 0032           81 IHOLD(II) = IBLNK1                                                     NUM01050
                     C                                                                          NUM01060
   ISN 0033          100 CONTINUE                                                               NUM01070
                     C                                                                          NUM01080
   ISN 0034              MOVSYM = KOUNT                                                         NUM01090
   ISN 0035              RETURN                                                                 NUM01100
   ISN 0036              END                                                                    NUM01110

                              *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0008  0009
II       0031  0032
KOL      0012  0013  0016  0025
CARD     0002  0005  0013
NDEX     0010  0016  0021  0022  0024  0030
ICHAR    0004  0006  0024
IHOLD    0004  0006  0009  0024  0032
ITEMP    0002  0003  0029
KOUNT    0011  0028  0028  0029  0034
IBLANK   0007  0014
IBLNK1   0004  0007  0009  0032
ICHAR4   0006  0007  0013  0014  0019
ICOMMA   0007  0019
IHOLD8   0003  0006  0029
MOVSYM   0002  0034

                              *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL    DEFINED    REFERENCES
     2      0009      0008
    10      0019      0014
    80      0028      0016  0019  0025
```



ORIGINAL PAGE IS
OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES |
|---|---|---|
| 81 | 0032 | 0031 |
| 100 | 0033 | 0012 0018 0022 0027 |

/ MOVSYM /  SIZE OF PROGRAM 000218 HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I SF | | I*4 | 000098 | II SF | | I*4 | 00009C | KOL SF | | I*4 | 0000A0 | CARD F | XR | I*4 | 000000 |
| NDEX SF | | I*4 | 0000A4 | ICHAR F | E | L*1 | 0000C0 | IHOLD S | E | L*1 | 0000C6 | ITEMP S | XR | A*8 | 000000 |
| KOUNT SF | | I*4 | 0000A8 | IBLANK | | I*4 | 0000AC | IBLNK1 F | | L*1 | 000094 | ICHAR4 S | E | I*4 | 0000C0 |
| ICOMMA | | I*4 | 000080 | IHOLD8 F | E | A*8 | 0000C8 | MOVSYM S | | I*4 | 000084 | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 9 | 00010A | 10 | 19 | 000156 | 80 | 28 | 00017E | 81 | 32 | 000198 |
| 100 | 33 | 0001AA | | | | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100001 | 2 | 0000E8 | 100002 | 10 | 000110 | 100003 | 13 | 000128 | 100004 | 16 | 000146 |
| 200001 | 17 | 000140 | 100005 | 18 | 000152 | 100006 | 21 | 000162 | 100007 | 24 | 00016A |
| 100008 | 27 | 00017A | 100010 | 34 | 000182 | | | | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*  SOURCE STATEMENTS = 35, PROGRAM SIZE = 536, SUBPROGRAM NAME =MOVSYM

*STATISTICS*  NO DIAGNOSTICS GENERATED

*****END OF COMPILATION *****  292K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.141/14.02.14        PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002                FUNCTION NUMBFL (CARD,XNMVEC)                                                  NUM00010
                     C                                                                                    NUM00020
                     C  MOVES FLOATING POINT NUMBERS FROM CARD TO ARRAY XNMVEC.                           NUM00030
                     C  STORES COUNT OF NUMBERS IN NUMB                                                   NUM00040
                     C                                                                                    NUM00050
   ISN 0003          C       IMPLICIT INTEGER (A-C)                                                       NUM00060
                     C                                                                                    NUM00070
   ISN 0004          C       DIMENSION CARD(1),XNMVEC(1)                                                  NUM00080
   ISN 0005          C       DATA    BLANK/' '/,COMMA/','/                                                NUM00090
   ISN 0006          C       DATA MINUS /'-'/                                                             NUM00100
   ISN 0007          C       DATA IZERO/'0'/,NINE/'9'/, KDOT/'.'/                                         NUM00110
                     C                                                                                    NUM00120
                     C  ITRIG = SWITCH FOR NUMBER COLLECTED AND NOT STORED                                NUM00130
   ISN 0008          C       ITRIG = 0                                                                    NUM00140
   ISN 0009          C       INEG = 1                                                                     NUM00150
   ISN 0010          C       XNUM = 0.                                                                    NUM00160
   ISN 0011          C       NDEX = 0                                                                     NUM00170
   ISN 0012          C        IDEC = 0                                                                    NUM00180
                     C                                                                                    NUM00190
   ISN 0013          C       DO 66  COL=1,80                                                              NUM00200
   ISN 0014          C       IF (CARD(COL) .EQ. KDOT) GO TO 20                                            NUM00210
   ISN 0016          C       IF (CARD(COL) .EQ. MINUS) GO TO 40                                           NUM00220
   ISN 0018          C       IF (CARD(COL) .EQ. BLANK) GO TO 60                                           NUM00230
   ISN 0020          C       IF (CARD(COL) .EQ. COMMA) GO TO 50                                           NUM00240
   ISN 0022          C       IF(CARD(COL) .LT. IZERO .OR. CARD(COL) .GT. NINE) GO TO 90                   NUM00250
   ISN 0024          C       CALL I4A1BN(CARD(COL),1,NWORD)                                               NUM00260
   ISN 0025          C       ITRIG=1                                                                      NUM00270
   ISN 0026          C       IF (IDEC .EQ. 1) GO TO 10                                                    NUM00280
   ISN 0028          C       XNUM = 10 * XNUM + NWORD                                                     NUM00290
   ISN 0029          C       GO TO 60                                                                     NUM00300
                     C                                                                                    NUM00310
                     C DECIMAL FRACTION                                                                   NUM00320
   ISN 0030        10        XWORD = NWORD                                                                NUM00330
   ISN 0031          C       XNUM = XNUM + XWORD/IPOWER                                                   NUM00340
   ISN 0032          C       IPOWER = IPOWER * 10                                                         NUM00350
   ISN 0033          C       GO TO 60                                                                     NUM00360
                     C                                                                                    NUM00370
                     C  DECIMAL                                                                           NUM00380
   ISN 0034        20        IDEC = 1                                                                     NUM00390
   ISN 0035          C       IPOWER = 10                                                                  NUM00400
   ISN 0036          C       GO TO 60                                                                     NUM00410
                     C                                                                                    NUM00420
                     C  MINUS SIGN                                                                        NUM00430
   ISN 0037        40        INEG = -1                                                                    NUM00440
   ISN 0038          C       GO TO 60                                                                     NUM00450
                     C                                                                                    NUM00460
                     C  END OF NUMBER. INEG IS SET TO 1 OR -1                                             NUM00470
   ISN 0039        50        NDEX = NDEX + 1                                                              NUM00480
   ISN 0040          C       XNMVEC(NDEX) = XNUM * INEG                                                   NUM00490
   ISN 0041          C       INEG = 1                                                                     NUM00500
   ISN 0042          C       IDEC = 0                                                                     NUM00510
   ISN 0043          C       ITRIG = 0                                                                    NUM00520
   ISN 0044          C       XNUM = 0.                                                                    NUM00530
   ISN 0045        60        CONTINUE                                                                     NUM00540
   ISN 0046        66        CONTINUE                                                                     NUM00550
                     C                                                                                    NUM00560
   ISN 0047          C    90 IF(ITRIG.EQ.0)GO TO 100                                                      NUM00570
   ISN 0049          C       NDEX = NDEX + 1                                                              NUM00580
   ISN 0050          C       XNMVEC(NDEX) = XNUM * INEG                                                   NUM00590
                     C                                                                                    NUM00600
   ISN 0051        100       NUMBFL = NDEX                                                                NUM00610
   ISN 0052          C       RETURN                                                                       NUM00620
   ISN 0053          C        END                                                                         NUM00630
                                                                                                          NUM00640

                          *****F O R T R A N    C R O S S    R E F E R E N C E    L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
COL      0013  0014  0016  0018  0020  0022  0022  0024
CARD     0002  0004  0014  0016  0018  0020  0022  0022  0024
IDEC     0012  0026  0034  0042
INEG     0009  0037  0040  0041  0050
KDOT     0007  0014
```

ORIGINAL PAGE IS
OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| SYMBOL | INTERNAL | STATEMENT NUMBERS | | | | | |
|---|---|---|---|---|---|---|---|
| NDEX | 0011 | 0039 | 0039 | 0040 | 0049 | 0050 | 0051 |
| NINE | 0007 | 0022 | | | | | |
| XNUM | 0010 | 0028 | 0028 | 0031 | 0031 | 0040 | 0044 | 0050 |
| BLANK | 0005 | 0018 | | | | | |
| COMMA | 0005 | 0020 | | | | | |
| ITRIG | 000A | 0025 | 0043 | 0047 | | | |
| IZERO | 0007 | 0022 | | | | | |
| MINUS | 0006 | 0016 | | | | | |
| NWORD | 0024 | 0028 | 0030 | | | | |
| XWORD | 0030 | 0031 | | | | | |
| IPOWER | 0031 | 0032 | 0032 | 0035 | | | |
| I4A1BN | 0024 | | | | | | |
| NUMBFL | 0002 | 0051 | | | | | |
| XNMVEC | 0002 | 0004 | 0040 | 0050 | | | |

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES | | | | |
|---|---|---|---|---|---|---|
| 10 | 0030 | 0026 | | | | |
| 20 | 0034 | 0014 | | | | |
| 40 | 0037 | 0016 | | | | |
| 50 | 0039 | 0020 | | | | |
| 60 | 0045 | 0018 | 0029 | 0033 | 0036 | 0038 |
| 66 | 0046 | 0013 | | | | |
| 90 | 0047 | 0024 | | | | |
| 100 | 0051 | 0047 | | | | |

/ NUMBFL / SIZE OF PROGRAM 00032E HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| COL | SFA | I*4 | 0000B4 | CARD | SFA XR | I*4 | 000000 | IDEC | S | I*4 | 0000B8 | INEG | SF | I*4 | 0000BC |
| KNOT | | I*4 | 0000C0 | NDEX | SF | I*4 | 0000C4 | NINE | | I*4 | 0000C8 | XNUM | SF | R*4 | 0000CC |
| BLANK | | I*4 | 0000D0 | COMMA | | I*4 | 0000D4 | ITRIG | S | I*4 | 0000D8 | IZERO | | I*4 | 0000DC |
| MINUS | | I*4 | 0000E0 | NWORD | SFA | I*4 | 0000E4 | XWORD | SF | R*4 | 0000E8 | IPOWER | SF | I*4 | 0000EC |
| I4A1BN | SF XF | | 000000 | NUMBFL | S | I*4 | 0000F0 | XNMVEC | S XR | R*4 | 000000 | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 30 | 000204 | 20 | 34 | 000248 | 40 | 37 | 000254 | 50 | 39 | 00025E |
| 60 | 45 | 000292 | 66 | 46 | 000292 | 90 | 47 | 00029E | 100 | 51 | 0002C8 |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100001 | 2 | 000114 | 100002 | 14 | 00014E | 100003 | 16 | 000160 | 100004 | 18 | 000172 |
| 100005 | 20 | 000184 | 100006 | 22 | 000196 | 200001 | 23 | 0001A8 | 100007 | 24 | 0001BA |
| 100008 | 28 | 0001DE | 100010 | 49 | 0002A4 | | | | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS* SOURCE STATEMENTS = 52, PROGRAM SIZE = 814, SUBPROGRAM NAME =NUMBFL

*STATISTICS* NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******  292K BYTES OF CORE NOT USED



ORIGINAL PAGE IS OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.141/14.17.41      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002              SUBROUTINE RDEXEC                                                          RDE00010
                C                                                                                   RDE00020
   ISN 0003             IMPLICIT INTEGER (A-X)                                                     RDE00030
                C                                                                                   RDE00040
                C  PURPOSE:  READ INFORMATION FROM THE EXEC FILE AND THE LARS DIRECTORY             RDE00050
                C            INFORMATION WRITTEN ON FILE 7 BY THE    PROGRAM                        RDE00060
                C                SEGMENT NUMBER                                                     RDE00070
                C                NUMBER OF ACQUISITONS                                              RDE00080
                C                ACQUISITION DATES                                                  RDE00090
                C                SUN ANGLES                                                         RDE00100
                C                LANDSAT NUMBERS                                                    RDE00110
                C                                                                                   RDE00120
                C                                                                                   RDE00130
   ISN 0004             COMMON /USER/ ICROP(4),ZX(4),SUNANG(8)                                     RDE00140
   ISN 0005             REAL*4 ZX                                                                  RDE00150
   ISN 0006             COMMON /MISC/ NOACQ, ACQDT(8), KSEGM, DATE(4),LANDST(8)                    RDE00160
   ISN 0007             REAL * 8 ACQDT                                                             RDE00170
                C                                                                                   RDE00180
                C  READ SEGMENT NUMBER                                                              RDE00190
                CCCC    WRITE (3,9999)                                                              RDE00200
   ISN 0008     9999    FORMAT (' ENTERING RDEXEC')                                                 RDE00210
   ISN 0009             READ (7,1000) KSEGM                                                        RDE00220
   ISN 0010     1000    FORMAT (20A4)                                                              RDE00230
                C                                                                                   RDE00240
                CCCC    WRITE (3,9998) KSEGM                                                        RDE00250
   ISN 0011     9998    FORMAT (' KSEGM=',A4)                                                      RDE00260
                C  READ NUMBER OF ACQUISITIONS                                                      RDE00270
   ISN 0012             READ(7,400) NOACQ                                                          RDE00280
   ISN 0013     400     FORMAT (2014)                                                              RDE00290
                C                                                                                   RDE00300
                C  READ ACQUISITION DATE                                                            RDE00310
   ISN 0014             READ (7,700) (ACQDT(I),I=1,NOACQ)                                          RDE00320
                CCCC    WRITE (3,700) (ACQDT(I),I=1,NOACQ)                                         RDE00330
   ISN 0015     700     FORMAT (10(2X,A5))                                                         RDE00340
                C                                                                                   RDE00350
                C  READ SUN ANGLES                                                                  RDE00360
   ISN 0016             READ (7,400) (SUNANG(I),I=1,NOACQ)                                         RDE00370
                C                                                                                   RDE00380
                C  READ LANDSAT NUMBERS                                                             RDE00390
   ISN 0017             READ (7,400) (LANDST(I),I=1,NOACQ)                                         RDE00400
                CCCC    WRITE (3,400) (LANDST(I),I=1,NOACQ)                                        RDE00410
                C                                                                                   RDE00420
   ISN 0018             RETURN                                                                     RDE00430
   ISN 0019             END                                                                        RDE00440

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL  INTERNAL STATEMENT NUMBERS
I       0014  0014  0014  0016  0016  0016  0017  0017  0017
ZX      0004  0005
DATE    0006
ACQDT   0006  0007  0014
ICROP   0004
KSEGM   0006  0009
NOACQ   0006  0012  0014  0016  0017
LANDST  0006  0017
RDEXEC  0002
SUNANG  0004  0016

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL   DEFINED   REFERENCES
   400     0013    0012  0016  0017
   700     0015    0014
  1000     0010    0009
  9998     0011
  9999     000A

                                                  /  RDEXEC /     SIZE OF PROGRAM 0001CA HEXADECIMAL BYTES

        NAME    TAG   TYPE  ADD.      NAME    TAG   TYPE  ADD.      NAME    TAG   TYPE  ADD.      NAME    TAG   TYPE  ADD.
        I  F          I*4   0000HC      ZX      C   R*4     NR      DATE    C     I*4     NR      ACQDT S C     R*8   000008
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
 ICROP      C  I*4   NR      KSEGM S    C  I*4  000048   NOACQ SF  C  I*4  000000   18COM# F  XF  I*4  000000
 LANDST S   C  I*4  00005C   RDEXEC        I*4  0000C0   SUNANG S  C  I*4  000020
```

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * USER* SIZE OF BLOCK 000040 HEXADECIMAL BYTES

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ICROP | I*4 | 000000 NR | ZX | R*4 | 000010 NR | SUNANG | I*4 | 000020 | | | |

NAME OF COMMON BLOCK * MISC* SIZE OF BLOCK 00007C HEXADECIMAL BYTES

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NOACQ | I*4 | 000000 | ACQDT | R*8 | 000008 | KSEGM | I*4 | 00004R | DATE | I*4 | 00004C NR |
| LANDST | I*4 | 00005C | | | | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR |
|---|---|---|
| 100001 | 2 | 0000DC |

FORMAT STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9999 | 8 | 000028 NR | 1000 | 10 | 00003C | 9998 | 11 | 000042 NR | 400 | 13 | 00004F |
| 700 | 15 | 000055 | | | | | | | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS* SOURCE STATEMENTS = 18, PROGRAM SIZE = 458, SUBPROGRAM NAME =RDEXEC

*STATISTICS* NO DIAGNOSTICS GENERATED

***** END OF COMPILATION ***** 296K BYTES OF CORE NOT USED

ORIGINAL PAGE IS OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.141/14.18.02      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

ISN 0002                SUBROUTINE RDFLES (ITYPE, HDR, VERTPX, VERTLN, IERROR)                          RDF00010
                  C     PURPOSE: MOVE DATA FROM A UNIVERSAL FORMAT TAPE TO  IDATA                       RDF00020
                  C        MOVE MULTIPLE CHANNELS OF DATA                                               RDF00030
                  CCC                                                                                   RDF00040
                  CCC      **********    INPUT    **********                                            RDF00050
                  C     DATA FILES ON UNITS 11 - 18                                                     RDF00060
                  CCC                                                                                   RDF00070
                  CCC      **********    OUTPUT   **********                                            RDF00080
                  C     ITYPE = 1  READ HEADER RECORDS AND POSITION TAPE FOR 1ST FILE                   RDF00090
                  C     ITYPE = 2  READ A LINE OF PIXELS INTO IDATA                                     RDF00100
                  CCC                 RADIANCE VALUES (196 X NUMBER OF CHANNELS)                        RDF00110
                  CCC                                                                                   RDF00120
ISN 0003                IMPLICIT INTEGER (A-Z)                                                          RDF00130
                  C                                                                                     RDF00140
ISN 0004                LOGICAL*1 HDR(1)                                                                RDF00150
ISN 0005                REAL*4 VERTPX(4), VERTLN(4)                                                     RDF00160
                  CCC                                                                                   RDF00170
                  C     IDATA HAS (72 + 4*196) CHARACTERS FOR A MAX OF 8 ACQUISITIONS                   RDF00180
ISN 0006                COMMON /LINES/ NXTLNE, LINENO, IDATA(856,8), NOACQS                             RDF00190
ISN 0007                LOGICAL*1 IDATA                                                                 RDF00200
                  CCC                                                                                   RDF00210
                  CCC   ITYPE=1, READ HEADER;  ITYPE=2, READ DATA                                       RDF00220
                                                                                                        RDF00230
                                                                                                        RDF00240
ISN 0008                IERROR = 0                                                                      RDF00250
ISN 0009          10    IF (ITYPE .EQ. 2) GO TO 100                                                     RDF00260
                  C                                                                                     RDF00270
                  C   SKIP OVER HEADER RECORDS                                                          RDF00280
ISN 0011                DO 15 I = 1,NOACQS                                                              RDF00290
ISN 0012                ITAPE = 10 + I                                                                  RDF00300
ISN 0013                REWIND ITAPE                                                                    RDF00310
ISN 0014                READ (ITAPE, 1111, END = 901) HDR                                               RDF00320
ISN 0015          1111  FORMAT (17(180A1))                                                              RDF00330
ISN 0016          15    CONTINUE                                                                        RDF00340
                  C                                                                                     RDF00350
ISN 0017                NOFLD=0                                                                         RDF00360
                  C*                                                                                    RDF00370
                  C   READ FIELD DEFINITION                                                             RDF00380
ISN 0018          20    CALL RDVFRT ( VERTPX, VERTLN, IERROR)                                           RDF00390
ISN 0019                SAMINC = 1                                                                      RDF00400
ISN 0020                LININC = 1                                                                      RDF00410
ISN 0021                WRITE (6,1600) SAMINC, LININC, (VERTPX(I),VERTLN(I),I=1,4)                      RDF00420
ISN 0022          1600  FORMAT(//, ' PIXEL SKIP FACTOR = ',I4, ' LINE SKIP FACTOR = ',I4,               RDF00430
                     1 /, ' VERTICES (PIXEL, LINE)=',5('(',F4.0,',',F4.0,')',1X)  )                     RDF00440
                  C*                                                                                    RDF00450
                  C     SET NEXT LINE NUMBER                                                            RDF00460
ISN 0023                NXTLNE = 1                                                                      RDF00470
ISN 0024                RETURN                                                                          RDF00480
                  C                                                                                     RDF00490
                  C  NO FILE                                                                            RDF00500
ISN 0025          901   IDISK = ITAPE - 10                                                              RDF00510
ISN 0026                WRITE (6,902) IDISK                                                             RDF00520
ISN 0027                WRITE (3,902) IDISK                                                             RDF00530
ISN 0028          902   FORMAT (' DISK FILE HEADER RECORD FOR FILE ',I2, ' IS MISSING.',                RDF00540
                     1 /, ' RUN WILL TERMINATE')                                                        RDF00550
ISN 0029                IERROR = 1                                                                      RDF00560
ISN 0030                RETURN                                                                          RDF00570
                  C                                                                                     RDF00580
                  CCC  READ LINES FROM RADIANCE VALUE FILES                                             RDF00590
                  C    SKIP OVER UNNEEDED LINES                                                         RDF00600
ISN 0031          100   IF (NXTLNE .EQ. LINENO) GO TO 120                                               RDF00610
                  C                                                                                     RDF00620
ISN 0033                ISKIP = LINENO - NXTLNE                                                         RDF00630
ISN 0034                DO 110 J = 1,NOACQS                                                             RDF00640
ISN 0035                ITAPE = 10 + J                                                                  RDF00650
ISN 0036                DO 110 I = 1,ISKIP                                                              RDF00660
ISN 0037                READ (ITAPE,1111, END=910) DUMMY                                                RDF00670
ISN 0038          110   CONTINUE                                                                        RDF00680
                  C                                                                                     RDF00690
ISN 0039          120   DO 130 I = 1,NOACQS                                                             RDF00700
ISN 0040                ITAPE = 10 + I                                                                  RDF00710
                  C   READ (72 + 4*196) CHARACTERS                                                      RDF00720
                                                                                                        RDF00730
```

ORIGINAL PAGE IS OF POOR QUALITY

```
ISN 0041          READ (ITAPE,1112,END=910) (IDATA(III,I),III=1,856)                RDF00740
ISN 0042    1112  FORMAT (9(100A1))                                                 RDF00750
ISN 0043    130   CONTINUE                                                          RDF00760
                  C                                                               RDF00770
ISN 0044          NXTLNE = LINENO + 1                                               RDF00780
                  C                                                               RDF00790
ISN 0045          RETURN                                                            RDF00800
                  C                                                               RDF00810
                  C  END OF FILE WHILE TRYING TO READ GRAY LEVEL RECORD             RDF00820
ISN 0046    910   IDISK = ITAPE -10                                                 RDF00830
ISN 0047          WRITE (3,911) NXTLNE, IDISK                                       RDF00840
ISN 0048          WRITE (6,911) NXTLNE, IDISK                                       RDF00850
ISN 0049    911   FORMAT (' END OF FILE WHILE TRYING TO READ RECORD',I4,            RDF00860
                 1  ' ON DISK FILE ', I4, /, ' RUN WILL TERMINATE')                 RDF00870
ISN 0050          IERROR = 1                                                        RDF00880
ISN 0051           RETURN                                                           RDF00890
                  C                                                               RDF00900
ISN 0052          END                                                               RDF00910
```

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| SYMBOL | INTERNAL STATEMENT NUMBERS | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| I | 0011 | 0012 | 0021 | 0021 | 0021 | 0021 | 0036 | 0039 | 0040 | 0041 |
| J | 0034 | 0035 | | | | | | | | |
| NDR | 0002 | 0004 | 0014 | | | | | | | |
| III | 0041 | 0041 | 0041 | | | | | | | |
| DUMMY | 0037 | | | | | | | | | |
| IDATA | 0006 | 0007 | 0041 | | | | | | | |
| IDISK | 0025 | 0026 | 0027 | 0046 | 0047 | 0048 | | | | |
| ISKIP | 0033 | 0036 | | | | | | | | |
| ITAPE | 0012 | 0013 | 0014 | 0025 | 0035 | 0037 | 0040 | 0041 | 0046 | |
| ITYPE | 0002 | 0009 | | | | | | | | |
| NOFLD | 0017 | | | | | | | | | |
| IERROR | 0002 | 0008 | 0018 | 0029 | 0050 | | | | | |
| LINENO | 0006 | 0031 | 0033 | 0044 | | | | | | |
| LININC | 0020 | 0021 | | | | | | | | |
| NOACQS | 0006 | 0011 | 0034 | 0039 | | | | | | |
| NXTLNE | 0006 | 0023 | 0031 | 0033 | 0044 | 0047 | 0048 | | | |
| RDFLES | 0002 | | | | | | | | | |
| RDVERT | 0018 | | | | | | | | | |
| SAMINC | 0019 | 0021 | | | | | | | | |
| VERTLN | 0002 | 0005 | 0018 | 0021 | | | | | | |
| VERTPX | 0002 | 0005 | 0018 | 0021 | | | | | | |

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES | |
|---|---|---|---|
| 10 | 0009 | | |
| 15 | 0016 | 0011 | |
| 20 | 0018 | | |
| 100 | 0031 | 0009 | |
| 110 | 0038 | 0034 | 0036 |
| *120 | 0039 | 0031 | |
| 130 | 0043 | 0039 | |
| 901 | 0025 | 0014 | |
| 902 | 0028 | 0026 | 0027 |
| 910 | 0046 | 0037 | 0041 |
| 911 | 0049 | 0047 | 0048 |
| 1111 | 0015 | 0014 | 0037 |
| 1112 | 0042 | 0041 | |
| 1600 | 0022 | 0021 | |

/ RDFLES / SIZE OF PROGRAM 00052A HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I | SF | I*4 | 0001BC | J | SF | I*4 | 0001C0 | NDR | S XR | L*1 | 000000 | III | F | I*4 | 0001C4 |
| DUMMY | S | I*4 | 0001C8 | IDATA | S C | L*1 | 000008 | IDISK | SF | I*4 | 0001CC | ISKIP | SF | I*4 | 0001D0 |
| ITAPE | SF | I*4 | 0001D4 | ITYPE | C | I*4 | 0001D8 | NOFLD | S | I*4 | 0001DC | IBCOM# | F XF | I*4 | 000000 |
| IERROR | SFA | I*4 | 0001E0 | LINENO | F C | I*4 | 000004 | LININC | SF | I*4 | 0001E4 | NOACQS | F C | I*4 | 0001AC8 |
| NXTLNE | SF C | I*4 | 000000 | RDFLES | | I*4 | 0001E8 | RDVERT | SF XF | | 000000 | SAMINC | SF | I*4 | 0001EC |
| VERTLN | SFA XR | R*4 | 000000 | VERTPX | SFA XR | R*4 | 000000 | | | | | | | | |

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * LINES* SIZE OF BLOCK 0001ACC HEXADECIMAL BYTES



ORIGINAL PAGE IS OF POOR QUALITY

| VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. | VAR. NAME | TYPE | REL. ADDR. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NXTLNF | I*4 | 000000 | LINENO | I*4 | 000004 | IDATA | L*1 | 000008 | NOACQS | I*4 | 000AC8 |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | | LABEL | ISN | ADDR | | LABEL | ISN | ADDR | | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 9 | 000248 | NR | 15 | 16 | 00029C | | 20 | 18 | 0002AE | NR | 901 | 25 | 000328 |
| 100 | 31 | 000380 | | 110 | 38 | 0003C4 | | 120 | 39 | 0003D8 | | 130 | 43 | 000416 |
| 910 | 46 | 00043A | | | | | | | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100001 | 2 | 000228 | 100002 | 11 | 000254 | 100003 | 12 | 000256 | 100004 | 14 | 00026C |
| 100005 | 14 | 000280 | 100006 | 17 | 0002A6 | 100007 | 21 | 0002E2 | 100008 | 21 | 000310 |
| 100009 | 33 | 000380 | 100010 | 35 | 00039A | 100011 | 37 | 0003A4 | 100012 | 37 | 000388 |
| 100013 | 39 | 0003CE | 100015 | 40 | 0003DA | 100016 | 41 | 0003E2 | 100017 | 41 | 0003F8 |
| 100020 | 44 | 000420 | | | | | | | | | |

FORMAT STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1111 | 15 | 000028 | 1600 | 22 | 000031 | 902 | 28 | 000095 | 1112 | 42 | 0000E1 |
| 911 | 49 | 0000EA | | | | | | | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*  SOURCE STATEMENTS =  51, PROGRAM SIZE =  1322, SUBPROGRAM NAME =RDFLES

*STATISTICS*  NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******  292K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)          OS/360  FORTRAN H EXTENDED          DATE 81.141/14.16.07      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002              SUBROUTINE RDVERT (VERTPX, VERTLN, IERROR)                                  RDF00920
                     C                                                                             RDF00930
                     C  PURPOSE--READ AND DECODE A CARD DESCRIBING VERTICES OF TRAINING OR         RDF00940
                     C           TEST FIELD.  CARD CAN BE ABSENT INDICATING COMPLETE               RDF00950
                     C           FIELD OR CARD CAN BE IN THE FOLLOWING FORMAT:                     RDF00960
                     C                                                                             RDF00970
                     C                (1,1), (X1,Y1), (X2,Y2), (X3,Y3), (X4,Y4)                    RDF00980
                     C                                                                             RDF00990
                     C           WHERE X1,Y1 IS THE UPPER LEFT CORNER AND THE VERTICES             RDF01000
                     C           CONTINUE IN A CLOCKWISE DIRECTION.                                RDF01010
                     C                                                                             RDF01020
   ISN 0003              IMPLICIT INTEGER(A-Z)                                                       RDF01030
   ISN 0004              INTEGER CARD(80)                                                            RDF01040
   ISN 0005              REAL*4 VERTPX(4), VERTLN(4)                                                 RDF01050
   ISN 0006              REAL*4 XDEFLT(4), YDEFLT(4)                                                 RDF01060
   ISN 0007              DATA BLANK/' '/, COMMA/','/,  OPNPAR/'('/,  CLSPAR/')'/                     RDF01070
   ISN 0008              DATA XDEFLT /1., 196., 196., 1./                                            RDF01080
   ISN 0009              DATA YDEFLT /1., 1., 117., 117./                                            RDF01090
                     C                                                                             RDF01100
                     C  READ CARD                                                                  RDF01110
   ISN 0010              READ(2, 6, END=990) CARD                                                    RDF01120
   ISN 0011           6  FORMAT(80A1)                                                               RDF01130
                     C                                                                             RDF01140
                     C  SET PROGRAM TO NO COLUMNS READ, NO VERTICES FOUND                          RDF01150
   ISN 0012              NOCOL=0                                                                     RDF01160
   ISN 0013              NOVERT = -1                                                                 RDF01170
                     C                                                                             RDF01180
                     C  SET SWITCH TO NO PIXEL NUMBER READ                                         RDF01190
   ISN 0014              IPIXFL = 0                                                                  RDF01200
                     C                                                                             RDF01210
                     C  FIRST NON-BLANK CHARACTER MUST BE A (                                      RDF01220
   ISN 0015           1  ICHAR=NXTCHR(CARD,NOCOL)                                                  RDF01230
   ISN 0016              IF(ICHAR.EQ.BLANK) GO TO 935                                                RDF01240
   ISN 0018              IF(ICHAR.NE.OPNPAR) GO TO 903                                               RDF01250
                     C                                                                             RDF01260
   ISN 0020           3  NUM=0                                                                      RDF01270
   ISN 0021           4    ICHAR = NXTCHR(CARD,NOCOL)                                              RDF01280
   ISN 0022              IF(ICHAR.EQ.BLANK) GO TO 935                                                RDF01290
                     C                                                                             RDF01300
                     C  CHECK FOR COMMA SEPARATING PIXEL, LINE                                     RDF01310
   ISN 0024              IF (ICHAR .EQ. COMMA) GO TO 7                                               RDF01320
                     C                                                                             RDF01330
                     C  CHECK FOR RIGHT PARENTHESIS FINISHING PIXEL,LINE COMPLET                   RDF01340
   ISN 0026              IF(CARD(NOCOL).EQ.CLSPAR, GO TO 8                                           RDF01350
                     C                                                                             RDF01360
                     C  CHARACTER MUST BE NUMERIC                                                  RDF01370
   ISN 0028              CALL I4AIAN(ICHAR,1,NUMCHR)                                                 RDF01380
   ISN 0029              IF((NUMCHR.LT.0).OR.(NUMCHR.GT.9)) GO TO 903                                RDF01390
                     C                                                                             RDF01400
                     C  NUMERIC CHARACTER, CONVERT TO COMPUTATIONAL NUMBER AND INCLUDE IN SUM      RDF01410
   ISN 0031              NUM=10*NUM+NUMCHR                                                           RDF01420
   ISN 0032              GO TO 4                                                                     RDF01430
                     C                                                                             RDF01440
                     C  VERTEX PIXEL NUMBER                                                        RDF01450
   ISN 0033           7  NOVERT = NOVERT + 1                                                       RDF01460
   ISN 0034              IF (NOVERT .GT. 4) GO TO 910                                                RDF01470
   ISN 0036              IF (IPIXFL .NE. 0 ) GO TO 903                                               RDF01480
   ISN 0038              IF (NUM .EQ. 0) GO TO 903                                                   RDF01490
   ISN 0040              IF (NOVERT .GT. 0) VERTPX(NOVERT) = NUM                                     RDF01500
   ISN 0042              IPIXEL = NUM                                                                RDF01510
   ISN 0043              GO TO 3                                                                     RDF01520
                     C                                                                             RDF01530
                     C  VERTEX LINE NUMBER                                                         RDF01540
   ISN 0044           8  IF (IPIXFL .EQ. 0) GO TO 903                                              RDF01550
   ISN 0046              IF (NUM .EQ. 0) GO TO 903                                                   RDF01560
   ISN 0048              IF (NOVERT .GT. 0) VERTLN(NOVERT) = NUM                                     RDF01570
                     C                                                                             RDF01580
                     C  CHECK FOR COMMA FOLLOWING (PIXEL,LINE)                                     RDF01590
   ISN 0050           10 ICHAR=NXTCHR(CARD,NOCOL)                                                  RDF01600
   ISN 0051              IF(ICHAR.EQ.BLANK) GO TO 930                                                RDF01610
   ISN 0053              IF(ICHAR.EQ.COMMA) GO TO 2                                                  RDF01620
                     C                                                                             RDF01630
                     C                                                                             RDF01640
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
                C  SOMETHING WRONG WITH ORDER OF CHARACTERS                                        RDFA1650
   ISN 0055       903 WRITE(6,904) CARD                                                           RDFA1660
   ISN 0056           WRITE (3,904) CARD                                                          RDFA1670
   ISN 0057       904 FORMAT(' ERROR IN VERTICES CARD',/,1X,80A1,/,' RUN WILL TERMINATE          RDFA1680
                     1 ')                                                                        RDFA1690
   ISN 0058           IERROR = 1                                                                  RDFA1700
   ISN 0059           RETURN                                                                      RDFA1710
                C                                                                                 RDFA1720
                C  TOO MANY VERTICES                                                              RDFA1730
   ISN 0060       910 WRITE (3,[illegible]) CARD                                                  RDFA1740
   ISN 0061           WRITE (6,[illegible]) CARD                                                  RDFA1750
   ISN 0062       911 FORMAT ('[illegible]XIMUM OF 4 VERTICES ALLOWED',/,1X,80A1,/,' RUN WILL TE RDFA1760
                     1RMINATE')                                                                  RDFA1770
   ISN 0063           IERROR                                                                      RDFA1780
   ISN 0064           RETURN                                                                      RDFA1790
                C                                                                                 RDFA1800
                C  END OF CARD                                                                    RDFA1810
   ISN 0065       930 IF (NOVERT .EQ. 4) RETURN                                                   RDFA1820
                C                                                                                 RDFA1830
                C  ERROR, CARD TERMINATES BEFORE 4TH VERTICES FINISHED                            RDFA1840
   ISN 0067       935 WRITE (6,936) CARD                                                          RDFA1850
   ISN 0068           WRITE (3,936) CARD                                                          RDFA1860
   ISN 0069       936 FORMAT (' END OF CARD BEFORE END OF 4TH SET OF VERTICES',/,1X,80A1,       RDFA1870
                     1 /,  ' RUN WILL TERMINATE')                                                RDFA1880
   ISN 0070           IERROR = 1                                                                  RDFA1890
   ISN 0071           RETURN                                                                      RDFA1900
                C                                                                                 RDFA1910
                C  NO FIELD DEFINITION CARD, DEFAULT TO WHOLE SCENE                               RDFA1920
   ISN 0072       990 DO 995 I = 1,4                                                              RDFA1930
   ISN 0073           VERTPX(I) = XDEFLT(I)                                                       RDFA1940
   ISN 0074           VERTLN(I) = YDEFLT(I)                                                       RDFA1950
   ISN 0075       995 CONTINUE                                                                    RDFA1960
   ISN 0076           RETURN                                                                      RDFA1970
                C                                                                                 RDFA1980
   ISN 0077           END                                                                         RDFA1990
```

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| SYMBOL | INTERNAL | STATEMENT NUMBERS | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I | 0072 | 0073 | 0073 | 0074 | 0074 | | | | | | | |
| NUM | 0020 | 0031 | 0031 | 0038 | 0040 | 0042 | 0046 | 0048 | | | | |
| CARD | 0004 | 0015 | 0015 | 0021 | 0026 | 0050 | 0055 | 0056 | 0060 | 0061 | 0067 | 0068 |
| BLANK | 0007 | 0022 | 0022 | 0051 | | | | | | | | |
| COMMA | 0007 | 0053 | 0053 | | | | | | | | | |
| ICHAR | 0015 | 0018 | 0018 | 0021 | 0022 | 0024 | 0028 | 0050 | 0051 | 0053 | | |
| NOCOL | 0012 | 0021 | 0021 | 0026 | 0050 | | | | | | | |
| CLSPAR | 0007 | 0026 | 0026 | | | | | | | | | |
| IERROR | 0002 | 0058 | 0063 | 0070 | | | | | | | | |
| IPIXEL | 0014 | 0036 | 0042 | 0044 | | | | | | | | |
| I4A1BN | 0028 | | | | | | | | | | | |
| NOVERT | 0013 | 0033 | 0033 | 0034 | 0040 | 0040 | 0048 | 0048 | 0065 | | | |
| NUMCHR | 0028 | 0029 | 0029 | 0031 | | | | | | | | |
| NXTCHR | 0015 | 0021 | 0050 | | | | | | | | | |
| OPNPAR | 0007 | 0018 | | | | | | | | | | |
| RDVERT | 0002 | | | | | | | | | | | |
| VERTLN | 0002 | 0005 | 0048 | 0074 | | | | | | | | |
| VERTPX | 0002 | 0005 | 0040 | 0073 | | | | | | | | |
| XDEFLT | 0006 | 0008 | 0073 | | | | | | | | | |
| YDEFLT | 0006 | 0009 | 0074 | | | | | | | | | |

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 0014 | 0053 | | | | | |
| 3 | 0028 | 0043 | | | | | |
| 4 | 0021 | 0032 | | | | | |
| 6 | 0011 | 0010 | | | | | |
| 7 | 0033 | 0024 | | | | | |
| 8 | 0044 | 0026 | | | | | |
| 10 | 0050 | | | | | | |
| 11 | 0015 | | | | | | |
| 903 | 0055 | 0018 | 0029 | 0036 | 0038 | 0044 | 0046 |
| 904 | 0057 | 0055 | 0056 | | | | |
| 910 | 0060 | 0034 | | | | | |
| 911 | 0062 | 0060 | 0061 | | | | |
| 930 | 0065 | 0051 | | | | | |
| 935 | 0067 | 0016 | 0022 | | | | |
| 936 | 0069 | 0067 | 0068 | | | | |



ORIGINAL PAGE IS OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES |
|---|---|---|
| 990 | 0072 | 0010 |
| 995 | 0075 | 0072 |

/ RDVERT / SIZE OF PROGRAM 0000CA HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I | SF | I*4 | 000184 | NUM | SF | I*4 | 000188 | CARD | SFA | I*4 | 000B8 | BLANK | | I*4 | 000180 |
| COMMA | | I*4 | 000190 | ICHAR | SFA | I*4 | 000194 | NOCOL | SFA | I*4 | 000198 | CLSPAR | | I*4 | 000190 |
| IBCOM# | F XF | R*4 | 000000 | IERRON | S | I*4 | 0001A0 | IPIXFL | S | I*4 | 0001A4 | [illegible] | SF XF | | [illegible] |
| NOVERT | SF | I*4 | 0001A8 | NUMCHR | SFA | I*4 | 0001AC | NXTCHR | F XF | I*4 | 000000 | OPNPAR | | I*4 | 000180 |
| RDVERT | | I*4 | 000184 | VERTLN | S XR | R*4 | 000000 | VERTPX | S XR | R*4 | 000000 | XDEFLT | F | R*4 | 0002F8 |
| YDEFLT | F | R*4 | 000308 | | | | | | | | | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 14 | 00038E | 11 | 15 | 000390 | NR | 3 | 20 | 0003AE | 4 | 21 | 0003B0 |
| 7 | 33 | 000424 | 8 | 44 | 000466 | | 10 | 50 | 000498 | 903 | 55 | 0004A6 |
| 910 | 60 | 0005IC | 930 | 65 | 000580 | | 935 | 67 | 0005A6 | 990 | 72 | 00060C |
| 995 | 75 | 000640 | | | | | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100001 | 2 | 000340 | 100002 | 10 | 000374 | 100003 | 18 | 0003A6 | 100004 | 24 | 0003C6 |
| 100005 | 26 | 0003CE | 100006 | 28 | 0003E2 | 100007 | 31 | 000414 | 100008 | 36 | 00042E |
| 100009 | 38 | 000434 | 100010 | 40 | 00043A | 100011 | 41 | 000440 | 100012 | 42 | 000460 |
| 100013 | 46 | 00046C | 100014 | 48 | 000472 | 100015 | 49 | 000478 | 100016 | 53 | 0004AE |
| 100017 | 66 | 000586 | 100018 | 73 | 000610 | 100019 | 76 | 00064A | | | |

FORMAT STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 11 | 000028 | 904 | 57 | 00002E | 911 | 62 | 000069 | 936 | 69 | 0000A8 |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS* SOURCE STATEMENTS = 76, PROGRAM SIZE = 1738, SUBPROGRAM NAME =RDVERT

*STATISTICS* NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 284K BYTES OF CORE NOT USED

*STATISTICS* NO DIAGNOSTICS THIS STEP



ORIGINAL PAGE IS OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.141/14.19.35      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002                SUBROUTINE RECT (P,Q,XX,XY,ICOR)                                        REC00016
                     C  PURPOSE:  DETERMINE IF POINT IS IN TRAINING FIELD                          REC00020
   ISN 0003                DIMENSION XX(4),XY(4),THETA(4),AM(4),AX(4),AY(4)                        REC00030
   ISN 0004                ICOR=1                                                                  REC00040
   ISN 0005                NN = 4                                                                  REC00050
                     C  CHECK IF POINT IF ONE OF BOUNDRY POINTS                                    REC00060
   ISN 0006                DO 4 M = 1,NN                                                           REC00070
   ISN 0007                IF (P.EQ.XX(M) .AND. Q.EQ.XY(M))  GO TO 5                               REC00080
   ISN 0009           4     CONTINUE                                                               REC00090
   ISN 0010                GO TO 6                                                                 REC00100
                     C  POINT IN BOUNDS                                                            REC00110
   ISN 0011           5    ICOR = 0                                                               REC00120
   ISN 0012                RETURN                                                                  REC00130
                     C  CHECK FOR POINT INSIDE OF BOUNDS                                           REC00140
   ISN 0013           6    DO 1 I = 1,NN                                                          REC00150
   ISN 0014                AX(I) = P - XX(I)                                                       REC00160
   ISN 0015                AY(I) = Q - XY(I)                                                       REC00170
   ISN 0016                XM=AX(I)*AX(I)+AY(I)*AY(I)                                              REC00180
   ISN 0017           1     AM(I) = SQRT(XM)                                                       REC00190
                     C                                                                             REC00200
   ISN 0018                DO 2 J=1,NN                                                             REC00210
   ISN 0019                K=J+1                                                                   REC00220
   ISN 0020                IF (K.GT.NN) K=1                                                        REC00230
   ISN 0022                XTH=(AX(J)*AX(K)+AY(K)*AY(J))/(AM(J)*AM(K))                             REC00240
   ISN 0023           2     THETA(J) = ACOS(XTH)                                                   REC00250
   ISN 0024                SUM = 0.                                                                REC00260
   ISN 0025                DO 3 L = 1,NN                                                           REC00270
   ISN 0026           3     SUM = SUM+THETA(L)                                                     REC00280
   ISN 0027                DIFF = SUM - 3.14159263*2.                                              REC00290
   ISN 0028                DIFF=ABS(DIFF)                                                          REC00300
   ISN 0029                IF (DIFF.LE.1.E-04) ICOR = 0                                            REC00310
   ISN 0031                RETURN                                                                  REC00320
   ISN 0032                END                                                                     REC00330

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0013  0014  0014  0015  0015  0016  0016  0016  0016  0017
J        0018  0019  0022  0022  0022  0023
K        0019  0020  0020  0022  0022  0022
L        0025  0026
M        0006  0007  0007
P        0002  0007  0014
Q        0002  0007  0015
AM       0003  0017  0022  0022
AX       0003  0014  0016  0016  0022  0022
AY       0003  0015  0016  0016  0022  0022
NN       0005  0006  0013  0018  0020  0025
XM       0016  0017
XX       0002  0003  0007  0014
XY       0002  0003  0007  0015
ABS      0028
SUM      0024  0026  0026  0027
XTH      0022  0023
ACOS     0021
DIFF     0027  0028  0028  0029
ICOR     0002  0004  0011  0029
RECT     0002
SQRT     0017
THETA    0003  0023  0026

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL    DEFINED    REFERENCES
     1      0017      0013
     2      0023      0018
     3      0026      0025
     4      0009      0006
     5      0011      0007
     6      0013      0010

                                  /    RECT /    SIZE OF PROGRAM 000396 HEXADECIMAL BYTES
```



ORIGINAL PAGE IS
OF POOR QUALITY

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I | SF | I*4 | 0000A4 | J | SF | I*4 | 0000A8 | K | SF | I*4 | 0000AC | L | SF | I*4 | 0000B0 |
| M | SF | I*4 | 0000B4 | P | F | R*4 | 0000B8 | Q | F | R*4 | 0000BC | AM | SF | R*4 | 0000C0 |
| AX | SF | R*4 | 0000EC | AY | SF | R*4 | 0000FC | NN | SF | I*4 | 000[illegible]0 | XM | SFA | R*4 | 0000C4 |
| XX | F XR | R*4 | 000000 | XY | F XR | R*4 | 000000 | SUM | SF | R*4 | 0006C8 | XTH | SFA | R*4 | 0000CC |
| ACOS | F XF | R*4 | 000000 | DIFF | SFA | R*4 | 0000D0 | IC0R | S | I*4 | 0000D4 | RECT | | R*4 | 0000D8 |
| SQRT | F XF | R*4 | 000000 | THETA | SF | R*4 | 00010C | | | | | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 9 | 000194 | 5 | 11 | 0001A0 | 6 | 13 | 0001C8 | 1 | 17 | 000230 |
| 2 | 21 | 00029E | 3 | 26 | 0002C4 | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 000001 | 2 | 000140 | 000002 | 7 | 000168 | 000001 | 8 | 00017E | 000003 | 10 | 00019C |
| 000004 | 14 | 0001CA | 000005 | 18 | 00024C | 000006 | 19 | 000248 | 000007 | 21 | 000258 |
| 000008 | 22 | 00025A | 000009 | 24 | 000288 | 000010 | 27 | 0002DE | 000011 | 30 | 0002FA |
| 000012 | 31 | 0002FE | | | | | | | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*  SOURCE STATEMENTS = 31, PROGRAM SIZE = 918, SUBPROGRAM NAME = RECT

*STATISTICS*  NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******  296K BYTES OF CORE NOT USED



ORIGINAL PAGE IS OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.141/14.21.07       PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002                SUBROUTINE SDEV(X,N,XB,S)                                          SDF00010
                  C  CALCULATES THE MEAN AND STANDARD DEVIATION OF A VECTOR                   SDF00020
                  CC                                                                          SDF00030
   ISN 0003               REAL*8 SS                                                          SDF00040
                  C  X        VECTOR                                                         SDF00050
                  C  N        NO. OF DATA POINTS                                             SDEA0060
                  C  XB       MEAN                                                           SDF00070
                  C  S        STANDARD DEVIATION                                             SDF00080
   ISN 0004               DIMENSION  X(1)                                                    SDF00090
   ISN 0005               XB=0.                                                              SDF00100
   ISN 0006               XN=N                                                               SDF00110
                  CCCC  WRITE (6,9999) (X(II),II=1,N)                                         SDF00120
                  CCCC 9999  FORMAT (' SDEV=',8F8.2,/,(       ',8F8.2))                       SDF00130
   ISN 0007               DO 1 I=1,N                                                         SDF00140
   ISN 0008     1         XB=XB+X(I)                                                         SDF00150
   ISN 0009               XB=XB/XN                                                           SDF00160
   ISN 0010               SS=0.                                                              SDF00170
   ISN 0011               DO 2 I=1,N                                                         SDF00180
   ISN 0012               XDIFF = X(I) - XB                                                  SDF00190
   ISN 0013     2          SS = SS + XDIFF * XDIFF                                           SDE00200
   ISN 0014                XN=XN-1                                                           SDF00210
   ISN 0015               XX = SS/XN                                                         SDF00220
   ISN 0016                 S=SQRT(XX)                                                       SDF00230
   ISN 0017                 RETURN                                                           SDF00240
   ISN 0018                 END                                                              SDF00250

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0007  0008  0011  0012
N        0002  0006  0007  0011
S        0002  0016
X        0002  0004  0008  0012
SS       0003  0010  0013  0013  0015
XB       0002  0005  0008  000A  0009  0009  0012
XN       0006  0009  0014  0014  0015
XX       0015  0016
SDEV     0002
SQRT     0016
XDIFF    0012  0013  0013

                          *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL   DEFINED   REFERENCES
     1      000A      0007
     2      0013      0011

                                                    /   SDEV /     SIZE OF PROGRAM 000206 HEXADECIMAL BYTES

      NAME      TAG   TYPE  ADD.      NAME      TAG   TYPE  ADD.      NAME      TAG   TYPE  ADD.      NAME      TAG   TYPE  ADD.
         I  SF         I*4   0000A4       N   F      I*4   0000A8       S  S       R*4   0000AC       X   F  XR  R*4   000000
        SS  SF         R*8   0000CA      XB  SF      R*4   0000B0      XN  SF       R*4   0000B4      XX  SFA     R*4   0000B8
      SDEV             R*4   0000HC    SQRT   F  XF  R*4   000000    XDIFF  SF       R*4   0000C0

  SOURCE STATEMENT LABELS

    LABEL  ISN    ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
        1    A  000114              2   13  000152

  COMPILER GENERATED LABELS

    LABEL  ISN    ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
   100001    7  0000E4         100002    9  00012E         100003   12  000140         100004   14  00016A

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I
*STATISTICS*     SOURCE STATEMENTS =      17, PROGRAM SIZE =      518, SUBPROGRAM NAME =  SDEV
*STATISTICS*  NO DIAGNOSTICS GENERATED
```



ORIGINAL PAGE IS OF POOR QUALITY

PAGE IS
OF POOR QUALITY



```
*LEVEL 2.3.0 (JUNE 78)      SHEV      OS/360  FORTRAN H EXTENDED      DATE 81.141/14.21.07
****** END OF COMPILATION ******
                                        296K BYTES OF CORE NOT USED
```

```
*LEVEL 2.3.0 (APR  78)                OS/360  FORTRAN H EXTENDED          DATE 81.139/13.30.32       PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002            SUBROUTINE TRJXM (X, Y, SIGMAY, NPTS, NTERMS, MODF, ESTIM, DELTAA,TR.J00010
                      1  SIGMAA, FLAMDA, YFIT, CHISQR, IND, WEIGHT, IPRINT)                TR.J00020
                     C                                                                       TR.J00030
                     C  PURPOSE:                                                             TR.J00040
                     C    MAKE A LEAST-SQUARE FIT TO NON-LINEAR FUNCTION FZNCTN              TR.J00050
                     C    AND ESTIMATE THE NON-LINEAR PARAMETER ESTIM                        TR.J00060
                     C                                                                       TR.J00070
                     C  REFERENCE:                                                           TR.J00080
                     C    D. W. MARQUARDT (1963) AN ALGORITHM FOR LEAST-SQUARES              TR.J00090
                     C    ESTIMATION OF NONLINEAR PARAMETERS, J. SOC. INDUST. APPLI. MATH.,  TR.J00100
                     C    11, P 431-441)                                                     TR.J00110
                     C                                                                       TR.J00120
                     C  INPUT:                                                               TR.J00130
                     C    X - TIME ARRAY AS DAYS OF YEAR / 100.                              TR.J00140
                     C    Y - ARRAY OF MEANS OF GREENNESS VALUES FOR TRAIN FLD (1 PER ACQ)   TR.J00150
                     C    SIGMAY - ARRAY OF CORRESPONDING ST. DEVS.                          TR.J00160
                     C    NPTS - NUMBER OF ACQUISITIONS                                      TR.J00170
                     C    NTERMS - 4                                                         TR.J00180
                     C    MODF - 1  (NOT USED, FORMERLY USED FOR WEIGHTING FACTOR)           TR.J00190
                     C    ESTIM - INITIAL GUESSES (CHANGED BY TRJXM)                         TR.J00200
                     C    FLAMDA -                                                           TR.J00210
                     C    WEIGHT -  ARRAY OF WEIGHTING VALUES CALC'D AS 1. / (SIGMAY**2)     TR.J00220
                     C    IPRINT -  USED IN DEBUG TO TURN PRINT STATEMENTS OFF AND ON        TR.J00230
                     C                                                                       TR.J00240
                     C  OUTPUT                                                               TR.J00250
                     C    ESTIM - ESTIMATED VALUES FITTING MODEL TO INPUT ARRAYS X, Y, SIGMAYTR.J00260
                     C    DELTAA - 1.E-5 ARRAY USED FOR NON-ANALYSTICAL FUNCTIONS            TR.J00270
                     C    SIGMAA - ARRAY OF ERROR ON APPROXIMATIONS ESTIM                    TR.J00280
                     C    FLAMDA -                                                           TR.J00290
                     C    YFIT -                                                             TR.J00300
                     C    CHISQR - CHI-SQUARE VALUE                                          TR.J00310
                     C                                                                       TR.J00320
   ISN 0003            REAL*8 ARRAY                                                        TR.J00330
   ISN 0004            REAL*8 WAIT, WEIGHT(8)                                              TR.J00340
                     C                                                                       TR.J00350
   ISN 0005            DIMENSION X(  (1),SIGMAY(1),ESTIM(1),DELTAA(1),SIGMAA(1),YFIT(1)    TR.J00360
   ISN 0006            DIMENSION ALP  ,10),BETA(10),DERIV(10),ARRAY(10,10),H(10)           TR.J00370
                     C                                                                       TR.J00380
                     C  CALC NUMBER OF DEGREES OF FREEDOM                                    TR.J00390
   ISN 0007            NDEF=NPTS-NTERMS                                                    TR.J00400
                     C                                                                       TR.J00410
                     C                                                                       TR.J00420
                     C  SET DELTAA TO VERY SMALL NUMBER                                      TR.J00430
   ISN 0008            DO 10 I = 1,NTERMS                                                  TR.J00440
   ISN 0009       10     DELTAA(I) = 1.E-4                                                  TR.J00450
                     C                                                                       TR.J00460
                     C  ZERO BETA AND ALPHA                                                  TR.J00470
   ISN 0010       31     DO 34 I=1,NTERMS                                                   TR.J00480
   ISN 0011            BETA(J)=0.                                                          TR.J00490
   ISN 0012            DO 34 K=1,J                                                         TR.J00500
   ISN 0013       34     ALPHA(J,K)=0.                                                      TR.J00510
                     C                                                                       TR.J00520
                     C  CALC ALPHA AND BETA FOR EACH ACQUISITION                             TR.J00530
   ISN 0014       41     DO 50 I=1,NPTS                                                     TR.J00540
                     C                                                                       TR.J00550
                     C                                                                       TR.J00560
                     CCCC    **** WARNING:  FERIV TO BE SUBSTITUTED FOR FZERVG               TR.J00570
   ISN 0015            CALL FZERVG(X,I,ESTIM,IND,DELTAA,NTERMS,DERIV,Y,SIGMAY,NPTS,WEIGHTTR.J00580
                      * )                                                                  TR.J00590
                     CCCC     CALL FERIV (X,Y,SIGMAY,I,ESTIM,DELTAA,NTERMS,DERIV,NPTS,WEIGHT)TR.J00600
                     C                                                                       TR.J00610
                     C  CALC. CHI SQ AS FUNCTION OF ESTIM                                    TR.J00620
                     C  ****  WARNING:  FNCTN TO BE SUBSTITUTED FOR FZNCTG                   TR.J00630
   ISN 0016       62     YFIT(I)=FZNCTG(X,I,ESTIM,IND)                                      TR.J00640
                     CCCC 62 YFIT(I) = FNCTN (X, Y, SIGMAY, I, ESTIM, NPTS, WEIGHT)        TR.J00650
                     C                                                                       TR.J00660
   ISN 0017            YXXX = Y(I) - YFIT(I)                                               TR.J00670
                     C                                                                       TR.J00680
   ISN 0018            IF (IPRINT .LT. 0.  .AND. I.EQ.1) WRITE (6,9998) I, ESTIM(I)        TR.J00690
   ISN 0019       9998 FORMAT (' I=',I2,' ESTIM(I)=',F12.6)                                  TR.J00700
                     C                                                                       TR.J00710
   ISN 0021            DO 46 I=1,NTERMS                                                    TR.J00720
   ISN 0022            WAIT = WEIGHT(I)                                                    TR.J00730
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0023              BETA(J)=BETA(J) + WAIT * YXXX * DERIV(J)                  TR J00740
ISN 0024              DO 46 K=1,J                                               TR J00750
ISN 0025            WAIT = WEIGHT(I)                                            TR J00760
ISN 0026      46      ALPHA(J,K)=ALPHA(J,K) + WAIT * DERIV(J) * DERIV(K)        TR J00770
ISN 0027      50      CONTINUE                                                  TR J00780
                      C                                                         TR J00790
ISN 0028      51      DO 53 J=1,NTERMS                                          TR J00800
ISN 0029              DO 53 K=1,J                                               TR J00810
ISN 0030      53      ALPHA(K,J)=ALPHA(J,K)                                     TR J00820
                      C                                                         TR J00830
ISN 0031      61      CHISQ1=FCHISQ (Y,SIGMAY,NPTS,NFREE,MODE,YFIT,WEIGHT)      TR J00840
                      C                                                         TR J00850
ISN 0032      71      DO 74 I=1,NTERMS                                          TR J00860
ISN 0033            ALPHJJ = ALPHA(J,J)                                         TR J00870
ISN 0034              DO 73 K=1,NTERMS                                          TR J00880
ISN 0035              ZR=ALPHJJ*ALPHA(K,K)                                      TR J00890
ISN 0036              IF(ZR.LE.0.0) ZR=1.E-25                                   TR J00900
ISN 0038              ZRAY=SQRT(ZR)                                             TR J00910
ISN 0039              IF(ZRAY.LE.1.E-25) ZRAY=1.E-25                            TR J00920
ISN 0041      73      ARRAY(J,K)=ALPHA(J,K)/ZRAY                                TR J00930
ISN 0042      74      ARRAY(J,J)=1.+FLAMDA                                      TR J00940
                      C                                                         TR J00950
ISN 0043      80      CALL MTINV1(ARRAY,NTERMS,DET)                             TR J00960
                      C                                                         TR J00970
ISN 0044      81      DO 84 J=1,NTERMS                                          TR J00980
ISN 0045            ALPHJJ = ALPHA(J,J)                                         TR J00990
ISN 0046              B(J)= ESTIM(J)                                            TR J01000
ISN 0047              DO 84 K=1,NTERMS                                          TR J01010
ISN 0048              ZR=ALPHJJ*ALPHA(K,K)                                      TR J01020
ISN 0049              IF(ZR.LE.0.0) ZR=1.E-25                                   TR J01030
ISN 0051              ZRAY=SQRT(ZR)                                             TR J01040
ISN 0052      84      B(J)=B(J)+BETA(K)*ARRAY(J,K)/ZRAY                         TR J01050
                      C                                                         TR J01060
                      C  CALC CHI SQ AS FUNCTION OF CALC. VALUE B               TR J01070
ISN 0053      91      DO 92 I=1,NPTS                                            TR J01080
                      CCCC   ****  WARNING:  FNCTN SUBSTITUTE FOR F2NCTG        TR J01090
ISN 0054      92      YFIT(I)=F2NCTG(X,I,B,IND)                                 TR J01100
                      CCCC 92YFIT(I) = FNCTN(X, Y, SIGMAY, I, ESTIM, NPTS, WEIGHT)  TR J01110
                      C                                                         TR J01120
ISN 0055      93      CHISQR=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT,WEIGHT)       TR J01130
ISN 0056              IF(CHISQ1-CHISQR)95,101,101                               TR J01140
ISN 0057      95      FLAMDA=10.*FLAMDA                                         TR J01150
ISN 0058              GO TO 71                                                  TR J01160
                      C                                                         TR J01170
ISN 0059      101     DO 103 J=1,NTERMS                                         TR J01180
ISN 0060              ESTIM(J)=B(J)                                             TR J01190
ISN 0061              ZR=ARRAY(J,J)/ALPHA(J,J)                                  TR J01200
ISN 0062              IF(ZR.LE.0.) ZR=1.E-25                                    TR J01210
ISN 0064      103     SIGMAA(J)=SQRT(ZR)                                        TR J01220
                      C                                                         TR J01230
ISN 0065              FLAMDA=FLAMDA * .1                                        TR J01240
ISN 0066      110     RETURN                                                    TR J01250
ISN 0067              END                                                       TR J01260
```

*******F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*******

| SYMBOL | INTERNAL STATEMENT NUMBERS | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B | 0006 | 0046 | 0052 | 0052 | 0054 | 0060 | | | | | | | | | | | | | |
| I | 0008 | 0009 | 0014 | 0015 | 0016 | 0016 | 0017 | 0017 | 0018 | 0018 | 0022 | 0025 | 0053 | 0054 | 0054 | | | | |
| J | 0010 | 0011 | 0012 | 0013 | 0021 | 0023 | 0023 | 0023 | 0024 | 0026 | 0026 | 0026 | 0028 | 0029 | 0030 | 0030 | 0032 | 0033 | 0033 |
| | 0041 | 0041 | 0042 | 0042 | 0044 | 0045 | 0045 | 0046 | 0046 | 0052 | 0052 | 0052 | 0059 | 0060 | 0060 | 0061 | 0061 | 0061 | 0061 |
| | 0064 | | | | | | | | | | | | | | | | | | |
| K | 0012 | 0013 | 0024 | 0026 | 0026 | 0026 | 0028 | 0030 | 0030 | 0034 | 0035 | 0035 | 0041 | 0041 | 0047 | 0048 | 0048 | 0052 | 0052 |
| X | 0002 | 0005 | 0015 | 0016 | 0054 | | | | | | | | | | | | | | |
| Y | 0002 | 0005 | 0015 | 0017 | 0031 | 0055 | | | | | | | | | | | | | |
| ZR | 0035 | 0036 | 0036 | 0038 | 0048 | 0049 | 0049 | 0051 | 0061 | 0062 | 0062 | 0064 | | | | | | | |
| DET | 0043 | | | | | | | | | | | | | | | | | | |
| IND | 0002 | 0015 | 0016 | 0054 | | | | | | | | | | | | | | | |
| BETA | 0006 | 0011 | 0023 | 0023 | 0052 | | | | | | | | | | | | | | |
| MODE | 0002 | 0031 | 0055 | | | | | | | | | | | | | | | | |
| NPTS | 0002 | 0007 | 0014 | 0015 | 0031 | 0053 | 0055 | | | | | | | | | | | | |
| SQRT | 0038 | 0051 | 0064 | | | | | | | | | | | | | | | | |
| WAIT | 0004 | 0022 | 0023 | 0025 | 0026 | | | | | | | | | | | | | | |
| YFIT | 0002 | 0005 | 0016 | 0017 | 0031 | 0054 | 0055 | | | | | | | | | | | | |
| YXXX | 0017 | 0021 | | | | | | | | | | | | | | | | | |
| ZRAY | 0038 | 0039 | 0039 | 0041 | 0051 | 0052 | | | | | | | | | | | | | |
| ALPHA | 0006 | 0011 | 0026 | 0026 | 0030 | 0030 | 0033 | 0035 | 0041 | 0045 | 0044 | 0061 | | | | | | | |
| ARRAY | 0003 | 0006 | 0041 | 0042 | 0041 | 0052 | 0051 | | | | | | | | | | | | |



ORIGINAL PAGE IS
OF POOR QUALITY

••••••F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G••••••

| SYMBOL | INTERNAL STATEMENT NUMBERS | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DERIV | 0004 | 0011 | 0021 | 0026 | 0026 | | | | | | | | |
| FSTIM | 0002 | 0005 | 0015 | 0016 | 001R | 0046 | 0060 | | | | | | |
| NFREE | 0007 | 0031 | 0055 | | | | | | | | | | |
| TRJXM | 0002 | | | | | | | | | | | | |
| ALPHJJ | 0033 | 0035 | 0045 | 004R | | | | | | | | | |
| CHISQR | 0002 | 0056 | 0056 | | | | | | | | | | |
| CHISQ1 | 0031 | 0056 | | | | | | | | | | | |
| DELTAA | 0002 | 0005 | 0009 | 0015 | | | | | | | | | |
| FCHISQ | 0031 | 0055 | | | | | | | | | | | |
| FLAMDA | 0002 | 0047 | 0057 | 0057 | 0065 | 0065 | | | | | | | |
| F2ERVG | 0015 | | | | | | | | | | | | |
| F2NCTG | 0016 | 0054 | | | | | | | | | | | |
| IPRINT | 0002 | 0018 | | | | | | | | | | | |
| NTINV1 | 0043 | | | | | | | | | | | | |
| NTERMS | 0002 | 0007 | 0009 | 0010 | 0015 | 0021 | 0028 | 0032 | 0034 | 0043 | 0044 | 0047 | 0059 |
| SIGMAA | 0002 | 0005 | 0064 | | | | | | | | | | |
| SIGMAY | 0002 | 0005 | 0015 | 0031 | 0055 | | | | | | | | |
| WEIGHT | 0002 | 0004 | 0015 | 0022 | 0025 | 0031 | 0055 | | | | | | |

••••••F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G••••••

| LABEL | DEFINED | REFERENCES | |
|---|---|---|---|
| 10 | 0009 | 0009 | |
| 31 | 0010 | | |
| 34 | 0013 | 0010 | 0012 |
| 41 | 0014 | | |
| 46 | 0026 | 0021 | 0024 |
| 50 | 0027 | 0014 | |
| 51 | 0028 | | |
| 53 | 0030 | 002R | 0029 |
| 62 | 0016 | | |
| 63 | 0031 | | |
| 71 | 0032 | 005R | |
| 73 | 0041 | 0034 | |
| 74 | 0042 | 0032 | |
| 80 | 0043 | | |
| 81 | 0044 | | |
| 84 | 0052 | 0044 | 0047 |
| 91 | 0053 | | |
| 92 | 0054 | 0053 | |
| 93 | 0055 | | |
| 95 | 0057 | 0054 | |
| 101 | 0059 | 0056 | 0056 |
| 103 | 0064 | 0059 | |
| 110 | 0066 | | |
| 999R | 0020 | 001R | |

/ TRJXM / SIZE OF PROGRAM 0000D3C HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| P | SFA | R*4 | 000198 | I | SFA | I*4 | 000144 | J | SF | I*4 | 000148 | K | SF | I*4 | 00014C |
| X | SFA XR | R*4 | 000000 | Y | SFA XR | R*4 | 000000 | ZR | SFA | R*4 | 000150 | DET | SFA | R*4 | 000154 |
| IND | SFA | I*4 | 000158 | BETA | SF | R*R | 000100 | NODE | FA | I*4 | 00015C | NPTS | SFA | I*4 | 000160 |
| SORT | F | R*4 | 000000 | WAIT | SF | R*R | 000190 | YFIT | SFA XR | R*4 | 000000 | YXXX | SF | R*4 | 000164 |
| ZMZY | SF | R*4 | 000164 | ALPHA | SF | R*4 | 000168 | AVRAY | SFA | R*R | 000180 | DERIV | SFA | R*4 | 000178 |
| FSTIM | SFA XR | R*4 | 000000 | NFREE | SFA | I*4 | 00016C | TRJXM | | R*4 | 000170 | ALPHJJ | SF | R*4 | 000174 |
| CHISQR | S | R*4 | 000178 | CHISQ1 | S | R*4 | 00017C | DELTAA | SFA XR | R*4 | 000000 | FCHISQ | F XF | R*4 | 000000 |
| FLAMDA | SF | R*4 | 000140 | F2ERVG | SF XF | | 000000 | F2NCTG | F XF | R*4 | 000000 | IRCOMM | F XF | I*4 | 000000 |
| IPRINT | | I*4 | 000184 | NTINV1 | SF XF | | 000000 | NTERMS | SFA | I*4 | 000188 | SIGMAA | S XR | R*4 | 000000 |
| SIGMAY | SFA XR | R*4 | 000000 | WEIGHT | SFA XR | R*R | 000000 | | | | | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | | LABEL | ISN | ADDR | | LABEL | ISN | ADDR | | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 9 | 000748 | | 41 | 14 | 000786 | | 34 | 13 | 000780 | | 41 | 14 | 0007A6 |
| 62 | 16 | 000744 | XR | 46 | 26 | 000BFA | | 50 | 27 | 000BF0 | | 51 | 29 | 000BF2 |
| 53 | 30 | 000AFA | | 63 | 31 | 000924 | | 71 | 32 | 000916 | | 73 | 41 | 000494 |
| 74 | 42 | 000AC8 | | 80 | 43 | 000BF4 | | 81 | 44 | 000AFE | XR | 84 | 52 | 000A62 |
| 91 | 53 | 000A8E | | 92 | 54 | 000AC2 | | 93 | 55 | 000AFF | | 95 | 57 | 000408 |
| 101 | 59 | 000818 | | 103 | 64 | 000454 | | 110 | 66 | 000A9A | XR | | | |

COMPILER GENERATED LABELS



ORIGINAL PAGE IS
OF POOR QUALITY

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10000 | 2 | 000720 | 10003 | 11 | 000770 | 10004 | 14 | 00079C | 10006 | 15 | 0007AA |
| 20000 | 19 | 000804 | 10007 | 19 | 00080C | 10008 | 21 | 000834 | 10009 | 22 | 000836 |
| 10010 | 25 | 000878 | 10011 | 27 | 000806 | 10014 | 29 | 0008F4 | 10015 | 31 | 00091E |
| 10017 | 33 | 000938 | 10018 | 35 | 00094E | 10019 | 37 | 00096C | 10020 | 38 | 000974 |
| 10021 | 40 | 000980 | 10024 | 45 | 000A00 | 10025 | 48 | 000A2E | 10026 | 50 | 000A4C |
| 10027 | 51 | 000A54 | 10028 | 53 | 000AH4 | 10031 | 60 | 000B1A | 10032 | 63 | 000B62 |

FORMAT STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9999 | 20 | 000B2A | | | | | | | | | |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS* SOURCE STATEMENTS = 66, PROGRAM SIZE = 3388, SUBPROGRAM NAME = TRJXM

*STATISTICS* NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******

272K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JAN 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.139/13.23.43       PAGE   1

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) ;INCORE(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EXPDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002                SUBROUTINE F2ERVG (X,I,ESTIM,IND,DELTAA,NTERMS,DERIV,Y,SIGMAY,    F2N00210
                         1 NPTS, WEIGHT)                                                     F2N00220
                     C                                                                       F2N00230
   ISN 0003                REAL*8 WAIT, WEIGHT(4)                                          F2N00240
                     C                                                                       F2N00250
   ISN 0004                DIMENSION X(1), ESTIM(1), DELTAA(1), DERIV(1), Y(1), SIGMAY(1)  F2N00260
                     C                                                                       F2N00270
   ISN 0005                ESTIM(1)=ABS(ESTIM(1))                                          F2N00280
                     C                                                                       F2N00290
   ISN 0006                ESTIM(4)=ABS(ESTIM(4))                                          F2N00300
   ISN 0007                XI=X(I)/ESTIM(4)                                                F2N00310
   ISN 0008                IF(XI.LE.1.) GO TO 1                                            F2N00320
                     C                                                                       F2N00330
                     CCCC  ****WARNING: FNCTN REPLACES F2NCTG                              F2N00340
   ISN 0010                RHO=F2NCTG(X, I, ESTIM, IND)                                    F2N00350
                     CCCC  RHO = FNCTN (X, Y, SIGMAY, I, ESTIM, NPTS, WEIGHT)              F2N00360
                     C                                                                       F2N00370
   ISN 0011                DERIV(1)=RHO/ ESTIM(1)                                          F2N00380
   ISN 0012                DERIV(2)=RHO * ALOG(XI)                                         F2N00390
   ISN 0013                DERIV(3)=RHO * ESTIM(4) *  ESTIM(4) * (1.-XI * XI)              F2N00400
   ISN 0014                DERIV(4)=RHO * (2. *  ESTIM(3) * ESTIM(4)- ESTIM(2)/ ESTIM(4)) F2N00410
   ISN 0015                RETURN                                                          F2N00420
   ISN 0016              1 DERIV(1)=1.0                                                    F2N00430
   ISN 0017                DERIV(2)=1.E-10                                                 F2N00440
   ISN 0018                DERIV(3)=1.E-10                                                 F2N00450
   ISN 0019                DERIV(4)=1.E-10                                                 F2N00460
   ISN 0020                RETURN                                                          F2N00470
   ISN 0021                END                                                             F2N00480


                        ******  O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G ******
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0002  0007  0010
X        0002  0004  0007  0010
Y        0002  0004
XI       0007  0008  0012  0013  0013
ABS      0005  0006
IND      0002  0010
RHO      0010  0011  0012  0014  0014
ALOG     0012
NPTS     0002
WAIT     0003
DERIV    0002  0006  0011  0012  0013  0014  0016  0017  0018  0019
ESTIM    0002  0004  0005  0005  0006  0006  0007  0010  0011  0013  0013  0014  0014  0014  0014
DELTAA   0002  0004
F2ERVG   0002
F2NCTG   0010
NTERMS   0002
SIGMAY   0002  0004
WEIGHT   0002  0003


                        ******  O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G ******
 LABEL   DEFINED   REFERENCES
     1      0016      0008

                                                    /  F2ERVG /     SIZE OF PROGRAM 0002E6 HEXADECIMAL BYTES

      NAME     TAG   TYPE  ADD.      NAME     TAG  TYPE  ADD.      NAME     TAG  TYPE  ADD.      NAME     TAG  TYPE  ADD.
         I  FA       I*4  0000AC         X  FA XR  R*4  000000         Y     XR  R*4  000000        XI  SFA      R*4  000080
       IND  FA       I*4  0000B4       PHO  SF     R*4  0000B8      ALOG  F  XF  R*4  000000      NPTS           I*4  0000BC
      WAIT           R*8  0000C8     DERIV  S  XR  R*4  000000    ESTIM  SFA XR  R*4  000000    DELTAA     XR  R*4  000000
   F2ERVG            R*4  0000C0   F2NCTG  F  XF  R*4  000000   NTERMS           I*4  0000C4   SIGMAY     XR  R*4  000000
    WEIGHT     XR     R*8  000000

 SOURCE STATEMENT LABELS

  LABEL  ISN  ADDR      LABEL  ISN  ADDR      LABEL  ISN  ADDR      LABEL  ISN  ADDR

  COMPILER GENERATED LABELS
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
   LABEL  ISN   ADDR      LABEL  ISN   ADDR      LABEL  ISN   ADDR      LABEL  ISN   ADDR
  100001    2  000110    100002   10  00014C
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE FREDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =      28, PROGRAM SIZE =      742, SUBPROGRAM NAME =F2EVG
*STATISTICS*  NO DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                    242K BYTES OF CORE NOT USED
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JAN  78)                  OS/360  FORTRAN H EXTENDED          DATE 81.139/13.23.45       PAGE   1

REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EXTDICT NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM BM FLAG(I)

   ISN 0002                FUNCTION FCHISQ (Y,SIGMAY,NPTS,NFREE,MODE,YFIT, WEIGHT)          F2N00490
                     C                                                                     F2N00500
                     C  PURPOSE:  DETERMINE THE GOODNESS OF FIT (CHI-SQUARE) OF THE DATA WITH F2N00510
                     C            THE FITTED FUNCTION                                      F2N00520
                     C                                                                     F2N00530
   ISN 0003                  REAL*8 CHISQ, WAIT, WEIGHT                                    F2N00540
                     C                                                                     F2N00550
   ISN 0004                  DIMENSION Y(1),  WEIGHT(1), SIGMAY(1), YFIT(1)                F2N00560
                     C                                                                     F2N00570
   ISN 0005                    CHISQ=0.                                                    F2N00580
                     C                                                                     F2N00590
   ISN 0006                  DO 50 I = 1,NPTS                                            F2N00600
                     CCCC      WEIGHT(I) = 1./SIGMAY(I)**2                               F2N00610
   ISN 0007                    AYF=Y(I)-YFIT(I)                                          F2N00620
   ISN 0008                    IF(ABS(AYF).LE.1.E-10) AYF=1.E-10                         F2N00630
   ISN 0010                    IF(ABS(AYF).GE.1.E+10) AYF=1.E+10                         F2N00640
   ISN 0012                  WAIT = WEIGHT(I)                                            F2N00650
   ISN 0013                    CHISQ=CHISQ+WAIT*AYF*AYF                                  F2N00660
   ISN 0014          50        CONTINUE                                                  F2N00670
                     C                                                                   F2N00680
   ISN 0015                    IF(CHISQ.LT.1.E-20) CHISQ=1.E-20                          F2N00690
   ISN 0017                    IF(CHISQ.GT.1.E+20) CHISQ=1.E+20                          F2N00700
   ISN 0019                    FREE=NFREE                                                F2N00710
   ISN 0020                    FCHISQ=CHISQ/FREE                                         F2N00720
                     C                                                                   F2N00730
   ISN 0021          40        RETURN                                                    F2N00740
   ISN 0022                    END                                                       F2N00750

                          ******F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G******
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0006  0007  0007  0012
Y        0002  0004  0007
ABS      0008  0010
AYF      0007  0008  0008  0010  0010  0013  0013
FREE     0019  0020
MODE     0002
NPTS     0002  0006
WAIT     0003  0012  0013
YFIT     0002  0004  0007
CHISQ    0003  0005  0013  0013  0015  0015  0017  0017  0020
NFREE    0002  0019
FCHISQ   0002  0020
SIGMAY   0002  0004
WEIGHT   0002  0003  0004  0012

                          ******F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G******
LABEL    DEFINED    REFERENCES
  40     0021
  50     0014       0006

                                                    /  FCHISQ /     SIZE OF PROGRAM 000280 HEXADECIMAL BYTES

      NAME     TAG   TYPE  ADD.      NAME     TAG   TYPE  ADD.      NAME     TAG   TYPE  ADD.      NAME     TAG   TYPE  ADD.
         I  SI        I*4  0000C0         Y  F  XR  R*4  000000       AYF  SFA     R*4  0000C4      FREE  SF        R*4  0000C8
      MODE            I*4  0000CC     NPTS  F      I*4  000010      WAIT  SF     R*4  0000E0      YFIT  F  XR  R*4  000000
    CHISQ  SF         R*8  0000F8    NFREE  F      I*4  000014    FCHISQ  S     R*4  000008     SIGMAY     XR  R*4  000000
    WEIGHT  F  XR     R*8  000000

 SOURCE STATEMENT LABELS

   LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR

      50   14  000196            40   21  0001DA NR

 COMPILER GENERATED LABELS

   LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
   100001    2  000110       100002    7  000136       100003    9  000154       100004   10  000160
   100005   11  00016E       100006   12  000176       100007   15  000198       100008   16  0001A5
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 74)        FCHISQ          OS/360 FORTRAN H EXTENDED          DATE 81.139/13.23.45        PAGE   2
   100009   17  000144          100010   18  000152          100011   19  000186
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOXSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =       21.  PROGRAM SIZE =       640.  SUBPROGRAM NAME =FCHISQ
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******

                                                    292K BYTES OF CORE NOT USED
```



ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.139/13.23.42      PAGE   1

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002              FUNCTION F2NCTG (X, I, ESTIM, IND)                                        F2N00010
                 C                                                                               F2N00020
                 C  PURPOSE:  F2NCTG = ESTIM(1) * (XI ** ESTIM(2)) * EXP(APM)                    F2N00030
                 C            WHERE APM = ESTIM(3) * ESTIM(4) * ESTIM(4) * (1. - XI * XI)        F2N00040
                 C                                                                               F2N00050
   ISN 0003              DIMENSION X(1), ESTIM(1)                                                  F2N00060
                 C                                                                               F2N00070
   ISN 0004              ESTIM(4)=ABS(ESTIM(4))                                                    F2N00080
   ISN 0005              ESTIM(1)=ABS(ESTIM(1))                                                    F2N00090
   ISN 0006              XI=X(1)/ESTIM(4)                                                          F2N00100
   ISN 0007              IF(XI.LE.1.) XI=1.                                                        F2N00110
                 C  ********** WARNING:  THIS LINE WAS COMMENTED OUT                              F2N00120
                 CCCC  IF(XI.GT.2.E) XI=2.E                                                      F2N00130
   ISN 0009              APM=ESTIM(3)*ESTIM(4)*ESTIM(4)*(1.-XI*XI)                                 F2N00140
   ISN 0010              IF(APM.LT.-25.) APM=-25.                                                  F2N00150
   ISN 0012              IF (ESTIM(2).GE.65.) ESTIM(2)=65.                                         F2N00160
   ISN 0014              IF (ESTIM(2).LE.-65.) ESTIM(2)=-65.                                       F2N00170
   ISN 0016              F2NCTG= ESTIM(1) * (XI ** ESTIM(2)) * EXP(APM)                            F2N00180
   ISN 0017              RETURN                                                                    F2N00190
   ISN 0018              END                                                                       F2N00200


                      *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
I        0002  0004
X        0002  0003  0006
XI       0004  0007  0007  0009  0009  0016
ABS      0004  0005
APM      0009  0010  0010  0016
EXP      0016
IND      0002
ESTIM    0002  0003  0004  0004  0005  0005  0006  0009  0009  0009  0012  0012  0014  0014  0016  0016
F2NCTG   0002  0018

                                                  /  F2NCTG /     SIZE OF PROGRAM 00021E HEXADECIMAL BYTES

      NAME     TAG   TYPE  ADD.        NAME     TAG   TYPE  ADD.        NAME     TAG   TYPE  ADD.        NAME     TAG   TYPE  ADD.
         I  F        I*4   0000A4          X  F  XR   R*4   000000          XI SF       R*4   0000AB         APM SFA      R*4   0000AC
       EXP  F  XF    R*4   000000        IND           I*4   000080       ESTIM SFA XR   R*4   000000      FRXPR8       8F   R*4   000000
    F2NCTG S         R*4   000084

  COMPILER GENERATED LABELS

   LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
   100001    2  0000E0        100002    8  000116        100003    9  000122        100004   11  00014C
   100005   12  000156        100006   13  000162        100007   14  00016A        100008   15  00017A
   100009   16  000184

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*      SOURCE STATEMENTS =      17, PROGRAM SIZE =      542, SUBPROGRAM NAME =F2NCTG

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                        292K BYTES OF CORE NOT USED
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED                DATE 81.139/13.26.24        PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

  ISN 0002              SUBROUTINE MTINV1(A,N,DET)                                     MTI00010
              C                                                                        MTI00020
              C         GAUSS REDUCTION INVERSION WITHOUT ROW AND COLUMN INTERCHANGE.  MTI00030
              C                                                                        MTI00040
  ISN 0003              REAL*8 A,PVT,T                                                 MTI00050
  ISN 0004              DIMENSION A(10,10)                                             MTI00060
  ISN 0005              DET=1.                                                         MTI00070
  ISN 0006              DO 1 J=1,N                                                     MTI00080
  ISN 0007              PVT=A(J,J)                                                     MTI00090
  ISN 0008              IF(DABS(PVT) .LT. 1.D-20) PVT=1.D-20                           MTI00100
  ISN 0010              DET=DET*PVT                                                    MTI00110
  ISN 0011              A(J,J)=1.0                                                     MTI00120
  ISN 0012              DO 2 K=1,N                                                     MTI00130
              C         DIVIDE THE PIVOT ROW BY PIVOT ELEMENT.                         MTI00140
  ISN 0013    2         A(J,K)=A(J,K)/PVT                                              MTI00150
              C         REDUCE THE NON-PIVOT ROWS.                                     MTI00160
  ISN 0014              DO 1 K=1,N                                                     MTI00170
  ISN 0015              IF(K-J) 3,1,3                                                  MTI00180
  ISN 0016    3         T=A(K,J)                                                       MTI00190
  ISN 0017              A(K,J)=0.0                                                     MTI00200
  ISN 0018              DO 4 L=1,N                                                     MTI00210
  ISN 0019    4         A(K,L)=A(K,L)-A(J,L)*T                                         MTI00220
  ISN 0020    1         CONTINUE                                                       MTI00230
  ISN 0021              RETURN                                                         MTI00240
  ISN 0022              END                                                            MTI00250

                   *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL  INTERNAL STATEMENT NUMBERS
A       0002 0003 0004 0007 0011 0013 0013 0016 0017 0019 0019 0019
J       0006 0007 0007 0011 0011 0013 0013 0015 0016 0017 0019
K       0012 0013 0013 0014 0015 0016 0017 0019 0019
L       0018 0019 0019 0019
N       0002 0006 0012 0014 0018
T       0003 0016 0019
DET     0002 0005 0010 0010
PVT     0003 0007 0008 0008 0010 0013
DABS    0008
MTINV1  0002

                   *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
LABEL   DEFINED   REFERENCES
  1     0020      0006 0014 0015
  2     0013      0012
  3     0016      0015 0015
  4     0019      0018

                                    /  MTINV1 /     SIZE OF PROGRAM 000254 HEXADECIMAL BYTES
   NAME   TAG  TYPE  ADD.         NAME  TAG  TYPE  ADD.         NAME  TAG  TYPE  ADD.         NAME  TAG  TYPE  ADD.
     A SF  XF  R*8   000000         J SF     I*4   000098          K SF    I*4   00009C          L SF    I*4   0000A0
     N F       I*4   0000A4         T SF     R*8   0000B0        DET SF    R*4   0000A8        PVT SFA   R*8   0000B8
 MTINV1        I*4   0000AC

  SOURCE STATEMENT LABELS

   LABEL  ISN   ADDR        LABEL  ISN   ADDR        LABEL  ISN   ADDR        LABEL  ISN   ADDR
     2     13  000130         3     16  000156         4     19  000192         1     20  0001D2

  COMPILER GENERATED LABELS

   LABEL  ISN   ADDR        LABEL  ISN   ADDR        LABEL  ISN   ADDR        LABEL  ISN   ADDR
  100001    2  0000CA      100002    7  0000EE      100003    9  000104      100004   10  00010C
  100005   14  00015C      100006   15  00015E      100009   21  0001DC      100019   21  0001E6
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(60) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*     SOURCE STATEMENTS =      21, PROGRAM SIZE =      596, SUBPROGRAM NAME = MTINV1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUL 78)  MAIN        OS/360  FORTRAN H EXTENDED        DATE 81.139/13.26.24        PAGE   2
*STATISTICS*  NO DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                     288K BYTES OF CORE NOT USED
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED          DATE 81.141/14.02.38      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002                FUNCTION NUMBR (CARD, NUMVEC)                                                  NUM00010
                     C                                                                                    NUM00020
                     C  SIMPLIFIED VERSION OF NUMBER                                                      NUM00030
                     C  MOVES NUMBERS FROM CARD TO ARRAY NUMVEC.                                          NUM00040
                     C  STORES COUNT OF NUMBERS IN NUMB                                                   NUM00050
                     C                                                                                    NUM00060
   ISN 0003                IMPLICIT INTEGER (A-Z)                                                         NUM00070
                     C                                                                                    NUM00080
   ISN 0004                DIMENSION CARD(1),NUMVEC(1)                                                    NUM00090
   ISN 0005                DATA    BLANK/' '/,COMMA/','/                                                  NUM00100
   ISN 0006                DATA MINUS /'-'/                                                               NUM00110
   ISN 0007                DATA IZERO/'0'/,NINE/'9'/                                                      NUM00120
                     C                                                                                    NUM00130
                     C  ITRIG = SWITCH FOR NUMBER COLLECTED AND NOT STORED                                NUM00140
   ISN 0008                ITRIG = 0                                                                      NUM00150
   ISN 0009                INEG = 1                                                                       NUM00160
   ISN 0010                NUM = 0                                                                        NUM00170
   ISN 0011                NDEX = 0                                                                       NUM00180
                     C                                                                                    NUM00190
   ISN 0012                DO 60  COL=1,80                                                                NUM00200
   ISN 0013                IF (CARD(COL).EQ.MINUS) GO TO 40                                               NUM00210
   ISN 0015                IF (CARD(COL).EQ.BLANK) GO TO 60                                               NUM00220
   ISN 0017                IF (CARD(COL).EQ.COMMA) GO TO 50                                               NUM00230
   ISN 0019                IF(CARD(COL).LT.IZERO.OR.CARD(COL).GT.NINE) GO TO 90                           NUM00240
   ISN 0021                CALL I4A1BN(CARD(COL),1,NWORD)                                                 NUM00250
   ISN 0022                NUM = 10 * NUM + NWORD                                                         NUM00260
   ISN 0023                ITRIG=1                                                                        NUM00270
   ISN 0024                GO TO 60                                                                       NUM00280
                     C                                                                                    NUM00290
                     C  MINUS SIGN                                                                        NUM00300
   ISN 0025          40      INEG = -1                                                                    NUM00310
   ISN 0026                  GO TO 60                                                                     NUM00320
                     C                                                                                    NUM00330
                     C  END OF NUMBER, INEG IS SET TO 1 OR -1                                             NUM00340
   ISN 0027          50      NDEX = NDEX + 1                                                              NUM00350
   ISN 0028                  NUMVEC(NDEX) = NUM * INEG                                                    NUM00360
   ISN 0029                  INEG = 1                                                                     NUM00370
   ISN 0030                  ITRIG = 0                                                                    NUM00380
   ISN 0031                  NUM = 0                                                                      NUM00390
   ISN 0032             60 CONTINUE                                                                       NUM00400
                     C                                                                                    NUM00410
   ISN 0033             90 IF(ITRIG.EQ.0)GO TO 100                                                       NUM00420
   ISN 0035                NDEX = NDEX + 1                                                              NUM00430
   ISN 0036                NUMVEC(NDEX) = NUM * INEG                                                    NUM00440
                     C                                                                                    NUM00450
   ISN 0037          100   NUMBR = NDEX                                                                   NUM00460
   ISN 0038                RETURN                                                                         NUM00470
   ISN 0039                END                                                                            NUM00480

                                  ******F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G******

SYMBOL   INTERNAL STATEMENT NUMBERS
COL      0012  0013  0015  0017  0019  0019  0021
NUM      0010  0022  0022  0028  0031  0036
CARD     0002  0004  0013  0015  0017  0019  0019  0021
INEG     0009  0025  0028  0029  0036
NDEX     0011  0027  0027  0028  0035  0035  0036  0037
NINE     0007  0019
BLANK    0005  0015
COMMA    0005  0017
ITRIG    000A  0023  0030  0033
IZERO    0007  0019
MINUS    0006  0013
NUMBR    0002  0037
NWORD    0021  0022
I4A1BN   0021
NUMVEC   0002  0004  0028  0036

                                  ******F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G******

LABEL    DEFINED    REFERENCES
  40       0025     0013
  50       0027     0017
```

ORIGINAL PAGE IS
OF POOR QUALITY

*****F O R T R A N  C R O S S  R E F E R E N C E  L I S T I N G*****

| LABEL | DEFINED | REFERENCES |
|---|---|---|
| 60 | 0032 | 0012 0015 0024 0026 |
| 90 | 0033 | 0019 |
| 100 | 0037 | 0033 |

/ NUMBR / SIZE OF PROGRAM 000252 HEXADECIMAL BYTES

| NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. | NAME | TAG | TYPE | ADD. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| COL | SFA | *4 | 00009C | NUM | SF | *4 | 0000A0 | CARD | SFA XR | *4 | 000000 | INEG | SF | *4 | 0000A4 |
| NDEX | SF | *4 | 0000A8 | NINE | | *4 | 0000AC | BLANK | | *4 | 0000B0 | COMMA | | *4 | 0000B4 |
| ITRIG | S | *4 | 0000B8 | IZERO | | *4 | 0000BC | MINUS | | *4 | 0000C0 | NUMBR | S | *4 | 0000C4 |
| NWOPD | SFA | *4 | 0000C8 | 14A1BN | SF XF | | 000000 | NUMVES | S XR | *4 | 000000 | | | | |

SOURCE STATEMENT LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 40 | 25 | 000190 | 50 | 27 | 0001A6 | 60 | 32 | 0001C4 | 90 | 33 | 0001D0 |
| 100 | 37 | 0001EC | | | | | | | | | |

COMPILER GENERATED LABELS

| LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR | LABEL | ISN | ADDR |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 100001 | 2 | 0000EC | 100002 | 13 | 000116 | 100003 | 15 | 000128 | 100004 | 17 | 00013A |
| 100005 | 19 | 00014C | 200001 | 20 | 00015E | 100006 | 21 | 000170 | 100008 | 35 | 0001D6 |

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

*STATISTICS*  SOURCE STATEMENTS = 38, PROGRAM SIZE = 594, SUBPROGRAM NAME = NUMBR

*STATISTICS*  NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ******  292K BYTES OF CORE NOT USED



ORIGINAL PAGE IS
OF POOR QUALITY

# APPENDIX E

PROGRAM INPUTS:
SEGMENT NUMBER: 123
ACQUISITION#S: 78107 78151 78197 78233 78259
SUN ANGLES 56 56 54 48 49
LANDSAT NUMBER 2 2 2 2 2



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

CROP INFORMATION SUMMARY
TYPE DATA

| CROP | | | | |
|---|---|---|---|---|
| GREE | CORN | | | |
| *END | 12.00. | 11.50. | 1.15. | 1.50 |

SUPPORTING RESEARCH MULTITEMPORAL CLASSIFICATION

PROCESSING DATE - 05/21/81 AT 13: 0

SEGMENT NUMBER - 123 CROP OF INTEREST - CORN

PIXEL SKIP FACTOR = 1 LINE SKIP FACTOR = 1
VERTICES (PIXEL, LINE)=( 58., 3.) ( 64., 3.) ( 66., 8.) ( 60., 8.) (

TRAINING FIELD - LINE NO. SAMP. NO.

| | |
|---|---|
| 3.0 | 58.0 |
| 3.0 | 64.0 |
| 8.0 | 66.0 |
| 8.0 | 60.0 |

MEANS AND STD. DEV. FOR TRAINING FIELD BASED ON 10 PIXELS -

| GREENES NUMBER | | ACQUISITION DATES 78107 | 78161 | 78197 | 78233 | 78269 | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | MEAN | 12.12 | 11.00 | 50.02 | 47.33 | 19.30 | 0.0 | 0.0 | 0.0 |
| | STD. DEV. | 1.10 | 1.01 | 4.31 | 2.40 | 1.73 | 0.0 | 0.0 | 0.0 |

CONSTANTS FOR MODEL -

| CHANNEL NUMBER | | A | ALPHA | BETA | TO | CHISQ |
|---|---|---|---|---|---|---|
| 1 | INITIAL | 12.00 | 11.50 | 1.15 | 1.50 | |
| | EST M( )= | 12.000000 | | | | |
| | EST M( )= | 12.056185 | | | | |
| | EST M( )= | 12.034371 | | | | |
| | EST M( )= | 11.9A74A3 | | | | |
| | EST M( )= | 12.110775 | | | | |
| | EST M( )= | 11.811844 | | | | |
| | EST M( )= | 11.865816 | | | | |
| | EST M( )= | 11.840574 | | | | |
| | EST M( )= | 11.83A770 | | | | |
| | EST M( )= | 11.837594 | | | | |
| | EST M( )= | 11.8374A0 | | | | |
| | FINAL | 11.84+- 0.00 | 19.94+- 0.00 | 2.10+- 0.00 | 1.61+- 0.00 | 2.87 |

PIXEL SKIP FACTOR = 1 LINE SKIP FACTOR = 1
VERTICES (PIXEL, LINE)=( 1., 1.) (196., 1.) (196.,117.) ( 1.,117.) (

| | |
|---|---|
| EST M( )= | 11.837478 |
| EST M( )= | 13.858064 |
| EST M( )= | 11.255501 |
| EST M( )= | 11.247550 |
| EST M( )= | 11.255669 |
| EST M( )= | 11.257948 |
| EST M( )= | 11.257960 |

CLASSIFICATION FILE - RUN TIME: CP= 42.52 VIRTUAL CP= 41.65 IN MINUTES

DIFFERENCE IN INITIAL, ENDING CP TIMES IN HUNDREDTHS SECS.=255145
DIFFERENCE IN INITIAL, ENDING VIRTUAL TIMES IN HUNDREDTHS SECS.=249916



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

SUMMARY OF ERRORS FOR THIS JOB  ERROR NUMBER  NUMBER OF ERRORS
207  3

NASA-JSC